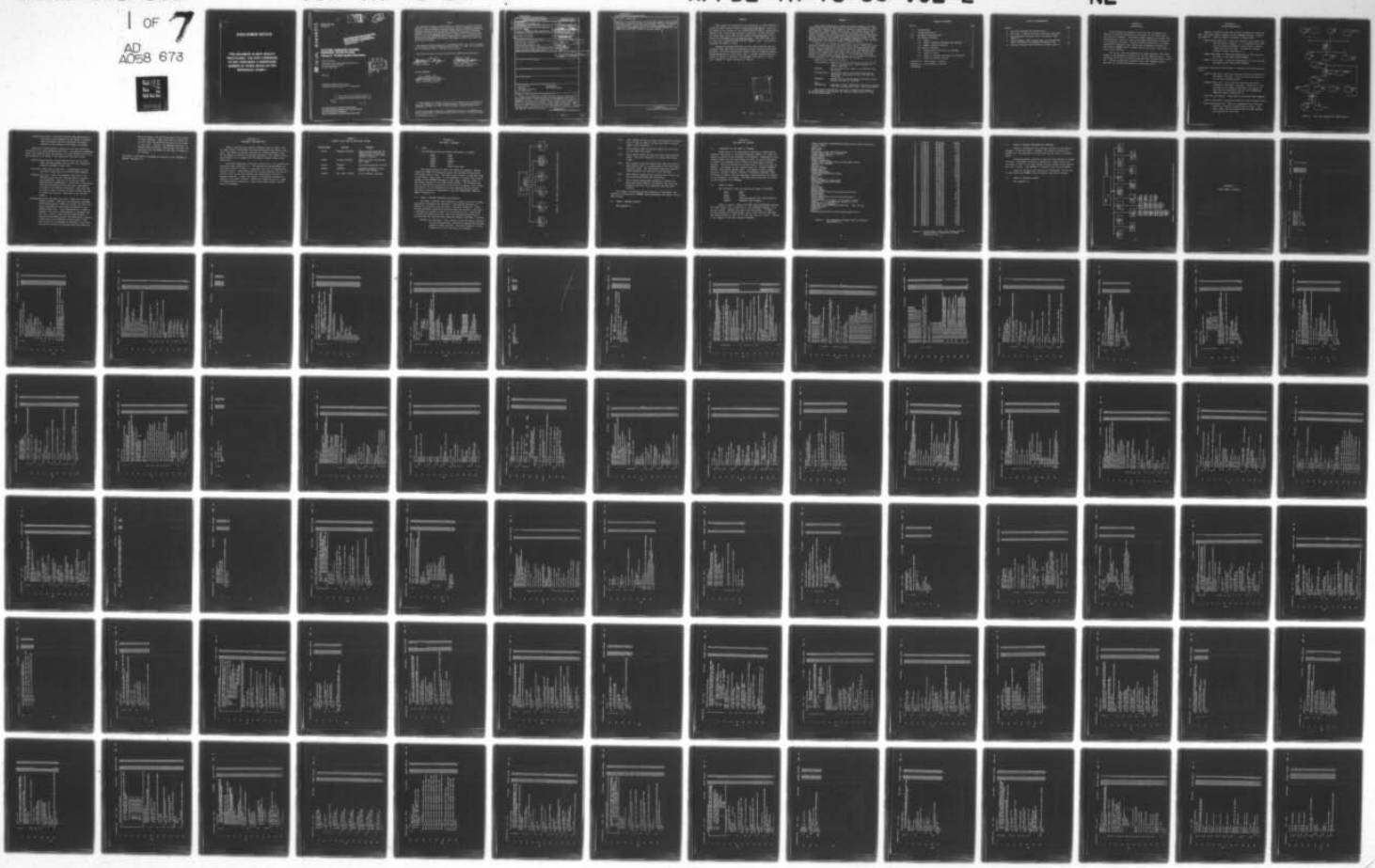
AD-A058 673 DAYTON UNIV OHIO RESEARCH INST F/G 9/2
PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). VOLUME II--ETC(U)
MAY 78 J G JENSEN F33615-76-C-3076
UNCLASSIFIED UDR-TR-78-34 AFFDL-TR-78-55-VOL-2 NL

1 OF 7
AD A058 673



DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

DDC FILE COPY, AD A058673

NH LEVEL II

B.S. 2

AFFDL-TR-78-55
Volume II

THIS DOCUMENT IS BEST QUALITY PRACTICABLE.
THE COPY FURNISHED TO DDC CONTAINED A
SIGNIFICANT NUMBER OF PAGES WHICH DO NOT
REPRODUCE LEGIBLY.

PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO)
Volume II: PLUGO System Description

JERRY G. JENSEN
UNIVERSITY OF DAYTON RESEARCH INSTITUTE
300 COLLEGE PARK AVENUE
DAYTON, OHIO 45469

DDC
RECEIVED
SEP 15 1978
F

MAY 1978

TECHNICAL REPORT AFFDL-TR-78-55
Final Report for Period November 1977 – February 1978

Approved for public release; distribution unlimited.

78 09 05 201

AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

James L. Terry

JAMES L. TERRY
Project Engineer

CHARLES J. COSENZA
Chief, Flight Vehicle Branch
Aeromechanics Division

FOR THE COMMANDER

ROBERT D. MCKELVEY, Col., USAF
Chief, Aeromechanics Division

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFFDL/FXS ,W-PAFB, OH 45433 to help us maintain a current mailing list".

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE/56780/28 July 1978 — 300

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AFFDL-TR-78-55-VOL-2	2. GOVT ACCESSION NO
	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). Volume II. PLUGO SYSTEM DESCRIPTION.	5. TYPE OF REPORT & PERIOD COVERED: Final Report. Nov 77 - Feb 78.
	UDR-TR-78-34
7. AUTHOR(s): Jerry G. Jensen	8. CONTRACT OR GRANT NUMBER(s): F33615-76-C-3076
9. PERFORMING ORGANIZATION NAME AND ADDRESS: University of Dayton Research Institute, 300 College Park Avenue, Dayton, Ohio 45469	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 2404-16-02
11. CONTROLLING OFFICE NAME AND ADDRESS: Flight Vehicle Branch, Aeromechanics Division, Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio 45433	12. REPORT DATE: May 78
	13. NUMBER OF PAGES: 592
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Control by Dayton University
Pg. 654 not at back

62201F

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Data Analysis
Plotting
Interactive Graphics
Data Reduction
Aerodynamics
Wind Tunnel Tests

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the function and use of a system of digital computer programs called PLUGO (Plotting Language Utilizing Graphical Optics). PLUGO combines batch processing with interactive graphics to provide the engineer with an analysis tool capable of handling large quantities of data. Basic input consists of a sequential file containing any number of fixed format data records. Data records contain values for up to 100 experimental variables. Output →

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

105 400

Block 20. (Continued)

consists of X-Y plots of user specified data subsets from the input data file. Plots are subject to inspection, editing, curve-fitting, and interpolation via an interactive graphics program. A special function creates trimmed plots of wind-tunnel generated force-moment data. Programs are written in CDC FORTRAN EXTENDED for execution on a CDC 6600 or CYBER 74 computer system.

PREFACE

This report was prepared by the University of Dayton Research Institute under Contract Number F33615-76-C-3067 EE 62. Work was performed at the request of the Flight Vehicle Branch, Aeromechanics Division, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio under Project Number 2404-16-02 during the period from November 1977 to February 1978. The author was Jerry G. Jensen with contributions from Dr. Duane G. Leet. The Project Engineer was James L. Terry.

Computer programs described in this report were written by personnel from the Digital Programming Branch, Digital Processing Division, Air Force Aeronautical Systems Division Computer Center, Wright-Patterson Air Force Base, Ohio. Systems analysis was performed by Thomas S. Rowland. Programmers were Sharon J. Foley, Capt. Stephen E. Hamm, and Wayne A. Jansen.

ACCESSION for
NTIS White Section
DDC Buff Section
UNANNOUNCED
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
Dist. SPECIAL
A 23 E.K

78 09 05 201

SUMMARY

This report describes a system of digital computer programs called PLUGO (Plotting Language Utilizing Graphical Optics). This report is contained in two volumes - Volume I, PLUGO User's Guide; and Volume II, PLUGO System Description. The PLUGO systems consists of two phases. Phase I is a batch program that scans the input data tape, captures data points, and creates plots from these points. Phase II is an interactive graphics program that permits the user to modify and analyze the plots created by Phase I. The PLUGO programs are written in CDC FORTRAN EXTENDED and are designed to execute on a CDC 6600 or CYBER 74 computer system.

Basic input to PLUGO is a sequential file containing any number of fixed format data records (data points). Each data point may contain floating-point values for up to 100 experimental variables. Output consists of X-Y plots, drawn on a Calcomp plotter of data subsets from the input data file. The user defines conditions on the experimental variables that determine how the data points are grouped into these subsets. The resulting plots are subject to modification through the use of an interactive-graphics program executed at a CDC CYBERGRAPHICS CRT terminal.

Modification functions include:

Curve-fitting - Data points on the plots may be curve-fitted using a straight-line, least squares, or spline algorithm.

Editing - Data points may be added to or deleted from the plots.

Interpolation - Data values may be calculated from the coefficients calculated by the curve-fitting functions.

Trimming - Trimmed plots of wind-tunnel generated force-moment data may be obtained.

Plot Information - Listings of plot information (data point values, curve-fit coefficients, etc.) may be obtained.

Thus PLUGO provides the user with a powerful yet simple to use data analysis tool capable of handling large amounts of data in an efficient manner.

TABLE OF CONTENTS

Section		Page
I	INTRODUCTION	1
II	SYSTEM DESCRIPTION	2
III	SOFTWARE CONSIDERATIONS	6
IV	THE PHASE I PROGRAM	8
	4.1 FILES	8
	4.2 PHASE I PROGRAM STRUCTURE AND FUNCTION	8
	4.3 PHASE I PROGRAM LISTING	10
V	THE PHASE II PROGRAM	11
	5.1 INITIATION OF THE PHASE II PROGRAM	11
	5.2 PHASE II FILES	11
	5.3 PHASE II PROGRAM STRUCTURE AND FUNCTION	14
	5.4 PHASE II PROGRAM LISTING	14
APPENDIX A:	PLUGO PHASE I LISTING	16
APPENDIX B:	PLUGO PHASE II LISTING	173
REFERENCES		586

LIST OF ILLUSTRATIONS

Figure		Page
1	Data Flow Through the PLUGO System	3
2	The Phase I Overlay Structure and Their Functions	9
3	The Cataloged Procedure Used to Initiate PLUGO Phase II	12
4	Record Number, Date, Time of Day, and CPU Seconds Used, as Stored in the PLUGO Accounting File	13
5	Overlay Structure and Functions of PLUGO Phase II	15

SECTION I
INTRODUCTION

This document is intended to provide the programmer with general information regarding the mechanics of the PLUGO Phase I and Phase II programs. Information about overlay structures, file communication, libraries, and non-standard programming techniques is included in the text. Specific remarks pertaining to variable usage and to specific logic used within individual routines are provided by COMMENT cards included within the Phase I and Phase II source code listings.

Due to the size and complexity of these programs, the PLUGO system is maintained by personnel at the ASD Computer Center. Thus questions pertaining to program modifications, malfunctions, status, etc., should be directed to Sharon Foley, ASD Computer Center, Building 676, Wright-Patterson Air Force Base, Ohio.

SECTION II
SYSTEM DESCRIPTION

Figure 1 illustrates the flow of data through the PLUGO system. The PLUGO system consists of two FORTRAN programs - Phase I and Phase II. Input to Phase I consists of three files -

OLDPL Source Code File - This file contains the source code for the Phase I program. It is suggested that this file be stored on magnetic tape or disc, however cards could also be used. It must be in OLDPL (UPDATE) format (Reference 1).

PLUGO Input Tape - A detailed description of this file may be found in the PLUGO USER'S MANUAL.

Phase I Input Deck - A detailed description of this file may be found in the PLUGO USER'S MANUAL.

Successful execution of PLUGO Phase I produces the following three files -

Phase I Plot Tape - This file contains the plots produced by the Phase I program. This tape may be plotted on an off-line Calcomp plotter.

Graphics Disc File - This file also contains the plots created by the Phase I program. This file is used as input to the Phase II program. The format of this file is consistant with the CDC INTERACTIVE GRAPHICS package (Reference 2).

Plot Data and Dayfile - Described in the PLUGO USER'S MANUAL.

Input to the Phase II program consists of the following files -

PLUGO Accounting File - This is a permanent disc file that contains a record of PLUGO Phase II usage. The date, time, and CPU seconds used for each session are recorded in this file.

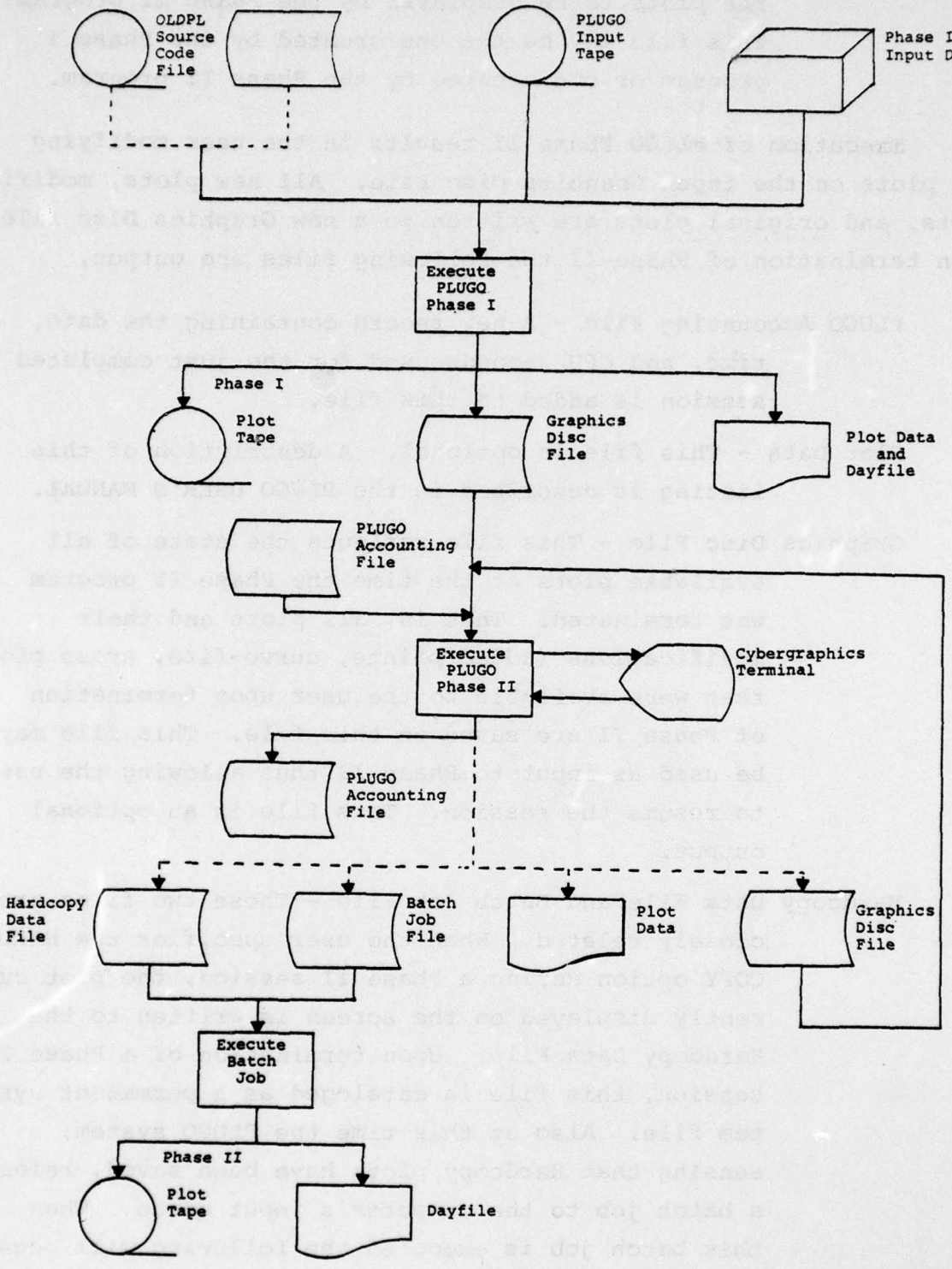


Figure 1. Data Flow Through the PLUGO System

Graphics Disc File - This file contains data pertaining to the plots to be displayed by the Phase II program. This file may be the one created by the Phase I program or one created by the Phase II program.

Execution of PLUGO Phase II results in the user modifying the plots on the input Graphics Disc File. All new plots, modified plots, and original plots are written to a new Graphics Disc file. Upon termination of Phase II the following files are output.

PLUGO Accounting File - A new record containing the date, time, and CPU seconds used for the just completed session is added to this file.

Plot Data - This file is optional. A description of this listing is described in the PLUGO USER'S MANUAL.

Graphics Disc File - This file reflects the state of all available plots at the time the Phase II program was terminated. That is, all plots and their modifications (added points, curve-fits, cross plots) that were available to the user upon termination of Phase II are saved on this file. This file may be used as input to Phase II thus allowing the user to resume the session. This file is an optional output.

Hardcopy Data File and Batch Job File - These two files are closely related. When the user specifies the HARDCOPY option during a Phase II session, the plot currently displayed on the screen is written to the Hardcopy Data File. Upon termination of a Phase II session, this file is cataloged as a permanent system file. Also at this time the PLUGO system, sensing that Hardcopy plots have been saved, releases a batch job to the computer's input queue. When this batch job is executed the following will occur. The job will attach the Hardcopy Data File that has

been cataloged. The specified Phase II Plot Tape is then requested. When this tape is mounted the job will copy the Hardcopy Data File to the Phase II Plot Tape. Finally, the Hardcopy Data File is purged from the system. The dayfile from this job signifies that the Phase II Plot Tape is ready to be plotted.

The Phase I and Phase II programs are written in CDC FORTRAN EXTENDED (Reference 3).

SECTION III
SOFTWARE CONSIDERATIONS

Table 1 contains the various libraries used by PLUGO. Two of these (IGS777 and CCPLOT) are used directly by the Phase I program. The Phase II program utilizes all of them; however, they do not appear in the CCL catalogued procedure. The reason is that the libraries are included with the Phase II object code in the PTASK file.

The PLUGO system was originally written for a 274 Interactive Graphics System (Reference 3). The system has been modified (update ident CYB777) to make it compatible with the 777 Interactive Graphics System. These modifications involve changing the initialization calls to the Data Manager and establishing an end-of-message character. PLUGO runs under a 274 emulator package contained in the IGS777 library. Therefore, PLUGO can run on a type 777 Cybergraphics terminal or, with a few modifications, on the older 274 terminal.

TABLE 1

LIBRARY FILES USED BY THE PLUGO SYSTEM

Library Name	Routine	Purpose
IGS777	Graphics routines	Used to build and read the Graphics Disc File and to display plots on the Cyber-graphics terminal.
CCPLOT	Calcomp routines	Used to create the Calcomp plot tapes.
CC6600	"STRING"	String manipulation routine.
NOSLIB	"PERMFIL"	Automatic permfile attach, extend, and return.
DRIVER	DMP, DMPX, RCOVER	Error recovery routines.

SECTION IV
THE PHASE I PROGRAM

4.1 FILES

The following files are used by the Phase I program.

FILE	TYPE
TAPE5	INPUT
TAPE6	OUTPUT
TAPE7	PLOT
TAPE3	PLUGO Input Tape

The record format for TAPE5 and TAPE6 is standard. Record format for TAPE7 is unnecessary since the Calcomp plotting software takes care of this. The TAPE3 record format is read off cards in the Phase I Input Deck. It is necessary to specify this record length in the PROGRAM statement of the Phase I program (refer to the Update Section in the PLUGO USER'S GUIDE). The Graphics Disc File is not directly specified in the Phase I program. Use of this file and how it is accessed by the Data Manager routines within the Phase I program is described in the CDC INTERACTIVE GRAPHICS MANUAL (Reference 2).

4.2 PHASE I PROGRAM STRUCTURE AND FUNCTION

The Phase I program uses overlays to reduce the central memory requirement necessary for execution. The program consists of a Nucleus (0,0 overlay) and six level 1 overlays (Figure 2). Each overlay performs a specific function within the program. The level 1 overlays are called in order ((1,0) through (6.0)). Each level 1 overlay is called only once, thus the following description of the overlays will describe the algorithm used by the Phase 1 program.

Nucleus (0,0 overlay) - This overlay drives the other overlays. It also contains error detection routines and routines common to other overlays. The main function of this overlay is to call the other six overlays.

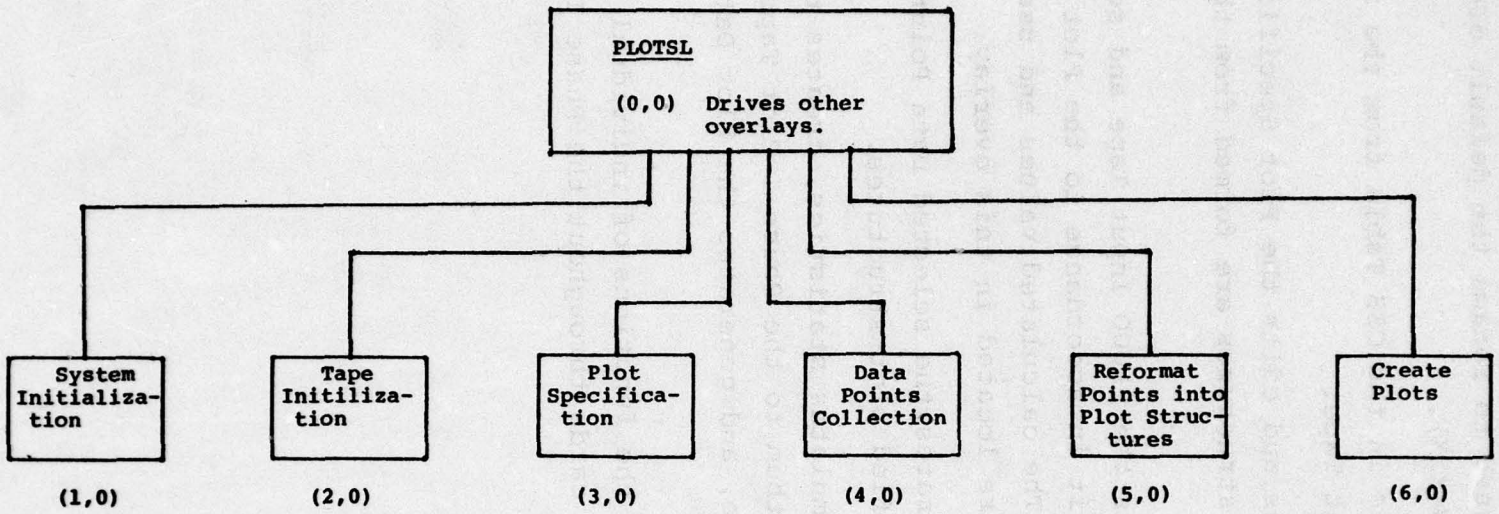


Figure 2. The Phase I Overlay Structure and Their Functions

(1,0) This overlay contains various initialization routines such as the one used to rename the default element names (OLD=V1 NEW=XXX).

(2,0) This overaly reads in the CSS Table from the front of the PLUGO Input Tape.

(3,0) This overlay reads and edits the Plot Specification cards. The plot structures are formed from these cards.

(4,0) This overlay reads the PLUGO Input Tape and selects Data Points from it in accordance to the Plot Specification cards. The calculated values and user defined functions are located in this overlay.

(5,0) This overlay separates the selected Data Points into the various specified plot structures.

(6.0) This overlay accumulates statistics, creates the plots and writes them to the Phase I Plot Tape and Graphics Disc File, and generates the Plot Data listing.

The usage of variables and the functions of individual subroutines is described by COMMENT cards throughout the Phase I source code listing.

4.3 PHASE I PROGRAM LISTING

See Appendix A.

SECTION V
THE PHASE II PROGRAM

5.1 INITIATION OF THE PHASE II PROGRAM

PLUGO Phase II is initiated by executing a Cyber Control Language (Reference 4) cataloged procedure (Figure 3). This procedure is stored on the permenent file PROCFIL. Depending on what problem number the user has logged in under, it may be necessary to attach the permanent file on which the cataloged procedure resides. This file should be attached using a local file name of PROCFIL. An example on the A system would be - ATTACH, PROCFIL, PROCFIL, ID=A730528, SN=ASD, MR=1. An example on the B system would be - ATTACH, PROCFIL,, PROCFIL, ID=D740020, SN=AFFDL, MR=1. The Phase II object code and necessary library modules are stored on the permanent file PTASK. The PLUGO Accounting File is stored on the permanent file PLUGOABS.

5.2 PHASE II FILES

The following files are used by the Phase II program:

FILE	TYPE
TAPE6	OUTPUT
TAPE43	PLUGO Accounting File (Input/Output)
TAPE99	HARDCOPY FILE (Output)

TAPE6 is used to output the PRINT PLOT INFORMATION listing. The TAPE6 buffer size is declared in the PROGRAM statement to be 64 octal words. TAPE43 also uses a 64 word buffer. A TAPE43 record contains the record number, date, time, and CPU time used. The record format is (1X, I5, 4X, A10, 5X, A10, 5X, F10.2). Figure 4 is a sample of the data stored in TAPE43.

```
.PROC,PLUGO,DMN,CY,DPB=A730528,TID=AB,FID=PLU,TAPE,TPB,HC=PLU.
IFE,SW1,JUMP3.
SWITCH,1.
ENDIF,JUMP3.
RFL,75500.
ATTACH,TASK,PTASK,ID=A730528,SN=ASD.
ATTACH,TEMPFL,DMN,*CY=CY,ID=DPB.
COPYBF,TEMPFL,DMFILE.
RETURN,TEMPFL.
REWIND,DMFILE.
REQUEST,TAPE99,*PF.
ATTACH,TAPE43,PLUGOABS,PW=SJF,ID=A730528,SN=ASD.
LIBLOAD(TASK,STARTER)
EXECUTE.
IFE,SW1,JUMP1.
REQUEST,NEW,*PF.
REWIND,DMFILE.
COPYBF,DMFILE,NEW.
CATALOG,NEW,DMN,RP=999,ID=DPB.
SKIP,JUMP1.
EDIT(S)
ENDIF,JUMP1.
RETURN,NEW,DMFILE,TAPE43,TASK.
CATALOG,TAPE99,HC,ID=A730528.
SKIP,JUMP2.
EXIT,S.
SKIP,NOPLOT.
ENDIF,JUMP2.
RETURN,TAPE99.
ROUTE,PLUGOHC,*TID=TID,*FID=FID,DC=IN,ST=CSB.
ENDIF,NOPLOT.
RETURN,TAPE99.
ROUTE,TAPE6,DC=PR,ST=CSA,*TID=TID,*FID=FID.
.DATA PLUGOHC
FID,CM10000,MT1,T10,ID200.  TPB, PLUGO CALCOMP
COMMENT.  * * * INTERCOM JOB, NO DECK * * * *
ATTACH,OLD,HC,ID=A730528.
LABEL,NEW,W,L=PLUGOHARD,VSN=TAPE,RING.  TPB, TTY JOB
COPYBF,OLD,NEW.
PURGE,OLD.
EXIT,U.
ROUTE,OUTPUT,DC=PR,*TID=TID,*FID=FID,DEF,ST=CSA.
```

Figure 3. The Cataloged Procedure Used to Initiate PLUGO Phase II

1	10/25/77	09.50.50.	22.95
2	11/01/77	09.12.38.	2.99
3	11/01/77	13.29.12.	10.07
4	11/02/77	09.45.27.	10.93
5	11/03/77	09.42.13.	1.37
6	11/08/77	10.48.05.	1.38
7	11/08/77	15.09.53.	7.64
8	11/10/77	12.46.50.	.34
9	11/10/77	13.19.17.	14.48
10	11/10/77	14.13.06.	2.43
11	11/11/77	09.27.31.	1.59
12	11/14/77	14.52.21.	72.28
13	11/14/77	16.14.08.	1.20
14	11/15/77	08.34.58.	4.65
15	11/16/77	09.4 .14.	1.90
16	11/16/77	10.16.59.	5.38
17	11/16/77	14.21.05.	43.20
18	11/16/77	15.26.11.	8.16
19	11/17/77	14.26.24.	29.33
20	11/17/77	14.32.43.	1.39
21	11/18/77	16.20.16.	35.28
22	11/21/77	10.08.11.	2.84
23	11/21/77	12.48.50.	6.59
24	11/22/77	09.42.28.	5.47
25	11/22/77	11.01.51.	3.00
26	11/22/77	14.30.02.	.36
27	11/23/77	13.57.44.	.39
28	11/28/77	09.27.54.	9.49
29	11/29/77	14.05.11.	.37
30	11/30/77	09.59.46.	.35
31	11/30/77	10.02.10.	.35
32	11/30/77	10.20.23.	9.58
33	12/01/77	12.55.54.	13.01
34	12/05/77	10.29.05.	7.12
35	12/09/77	16.07.41.	36.72
36	12/13/77	13.17.20.	11.32
37	12/15/77	09.34.56.	.32
38	12/19/77	09.23.28.	2.56
39	12/19/77	09.53.45.	4.34
40	12/19/77	10.21.45.	.32
41	12/19/77	10.35.07.	2.66
42	12/19/77	10.45.40.	9.06
43	12/20/77	15.16.37.	1.81
44	12/20/77	15.43.41.	11.30
45	12/21/77	09.06.03.	9.39
46	12/21/77	09.17.55.	5.06
47	12/21/77	13.42.39.	12.56
48	12/21/77	14.09.10.	3.15
49	12/21/77	14.16.09.	5.82
50	12/21/77	15.30.55.	12.72
51	12/22/77	10.03.02.	5.20
52	12/22/77	10.44.35.	.33

Figure 4. Record Number, Date, Time of Day, and CPU Seconds Used, as Stored in the PLUGO Accounting File

5.3 PHASE II PROGRAM STRUCTURE AND FUNCTION

Figure 5 illustrates the overlay structure of the Phase II program. Unlike the Phase I program, the Phase II overlays are loaded into core depending on which function the user is currently using.

Transfer between overlays is done with the AETSKC and AERTRN routines. The AETSKC routines is similar to a CALL function but is unique in that it can call another overlay without going through the (0,0) overlay. The AERTRN is a RETURN to the calling AETSKC.

Usage of variables and functions of individual subroutines is described with COMMENT cards throughout the source listing.

5.4 PHASE II PROGRAM LISTING

See Appendix B.

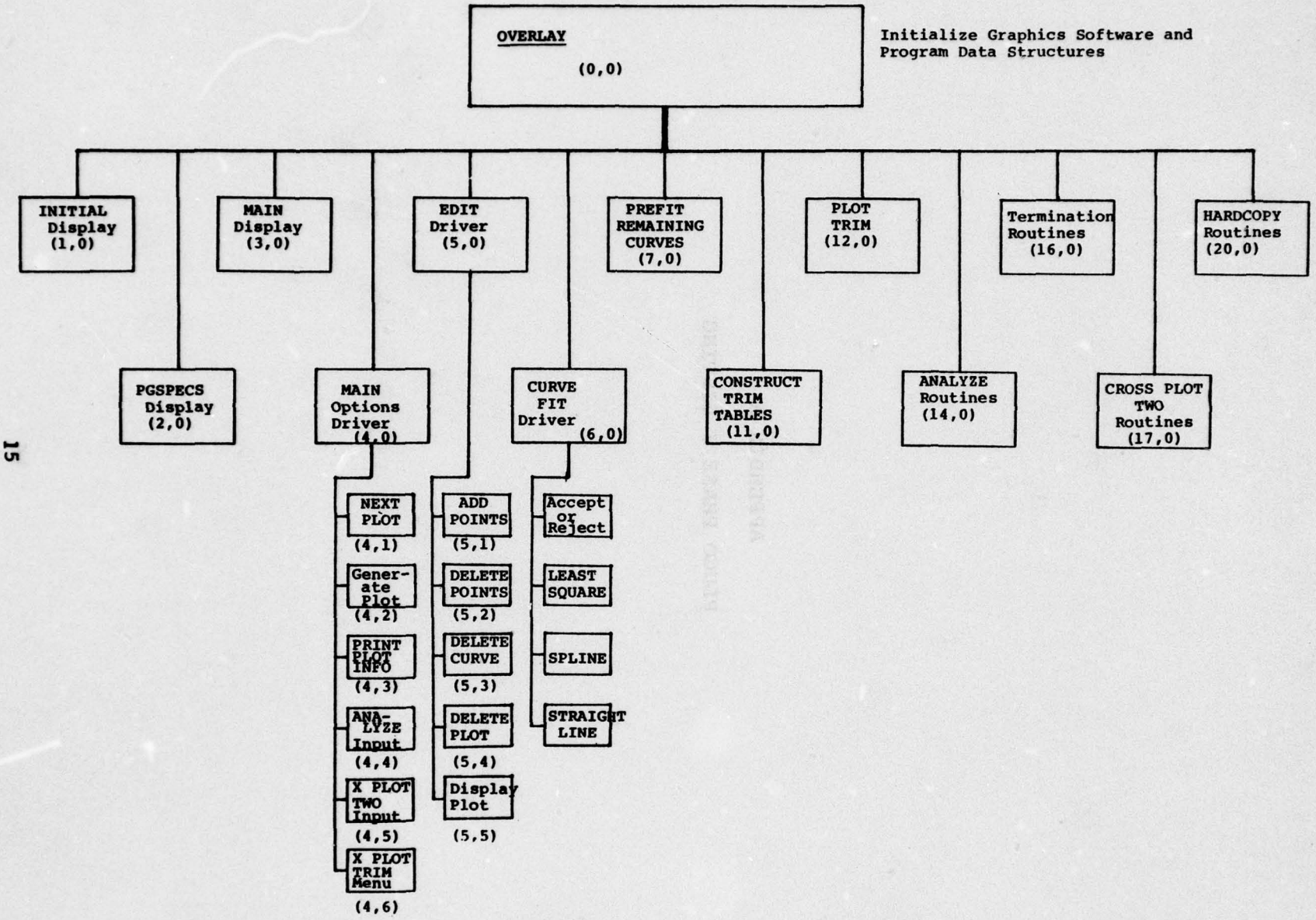


Figure 5. Overlay Structure and Functions of PLUGO Phase II.

APPENDIX A

PLUGO PHASE I LISTING

```
      OVERLAY(PLOTSL,0,0)                                                    DRIVER     2
      SUBROUTINE ERROR78                                                     DRIVER     3
      LOGICAL CHECK                                                          DRIVER     4
      COMMON/ERROR/CHECK                                                     DRIVER     5
C                                                                            DRIVER     6
C     SUBROUTINE RECOVERS FROM ERROR 78  SEE FORTRAN MANUAL G - 10           DRIVER     7
C                                                                            DRIVER     8
      CHECK = .TRUE.                                                         DRIVER     9
      RETURN                                                                 DRIVER    10
      END                                                                    DRIVER    11
```

```
      SUBROUTINE ERRORM(FMT,ISWTCH,NAMSUB)                                  DRIVER     12
C                                                                           DRIVER     13
C        THIS ROUTINE OUTPUTS SYSTEM ERROR MESSAGES                         DRIVER     14
C                                                                           DRIVER     15
      INTEGER FMT,CHRPOS                                                    DRIVER     16
      LOGICAL FATAL                                                         DRIVER     17
      COMMON /LASTPOS/ CHRPOS                                               DRIVER     18
      DATA FATAL /.FALSE./                                                  DRIVER     19
      IF(ISWTCH .LT. 0) FATAL=.TRUE.                                        DRIVER     20
      ISWTCH=IABS(ISWTCH)                                                   DRIVER     21
      IPOS=CHRPOS-1                                                         DRIVER     22
      GO TO (10,20,30),ISWTCH                                               DRIVER     23
   10 WRITE(6,1000) IPOS,NAMSUB                                             DRIVER     24
      GO TO 100                                                             DRIVER     25
   20 WRITE(6,1001) IPOS,NAMSUB                                             DRIVER     26
      GO TO 100                                                             DRIVER     27
   30 WRITE(6,1002) NAMSUB                                                  DRIVER     28
C                                                                           DRIVER     29
C     OUTPUT MESSAGE                                                        DRIVER     30
C                                                                           DRIVER     31
  100 CONTINUE                                                              DRIVER     32
      WRITE(6,FMT)                                                          DRIVER     33
      WRITE(6,1004)                                                         DRIVER     34
 1004 FORMAT(///)                                                           DRIVER     35
      IF(FATAL) GO TO 200                                                   DRIVER     36
      RETURN                                                                DRIVER     37
  200 CONTINUE                                                              DRIVER     38
      WRITE(6,1003)                                                         DRIVER     39
      STOP                                                                  DRIVER     40
 1000 FORMAT(//,30H ERROR DETECTED BEFORE COLUMN ,I2,15H BY SUBROUTINE ,    DRIVER     41
     1  A10,15H RECORD DELETED)                                            DRIVER     42
 1001 FORMAT(//,30H ERROR DETECTED BEFORE COLUMN ,I2,15H BY SUBROUTINE ,    DRIVER     43
     1  A10,17H ARGUMENT DELETED)                                          DRIVER     44
 1002 FORMAT(//,30H ERROR DETECTED BY SUBROUTINE ,A10)                      DRIVER     45
 1003 FORMAT(//,16H **FATAL ERROR**,//)                                     DRIVER     46
      END                                                                   DRIVER     47
```

```
      1                PROGRAM MAIN(INPUT,OUTPUT,PLOT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE3,      DRIVER     48
                      X TAPE7=PLOT)                                                       DRIVER     49
                       EXTERNAL ERROR78                                                   DRIVER     50
                       LOGICAL LISTSF,POINTS,ROCHK                                        DRIVER     51
      5                INTEGER ELETAB,FUNCTB,SURFTB,TAPENO,UNITNO                         DRIVER     52
                       INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50    DRIVER     53
                       COMMON /CONSTN/ BOVERC,NOOFOG                                      DRIVER     54
                       COMMON /ERROR/ ROCHK                                               DRIVER     55
                       COMMON/HEADPTR/IHEADR                                              DRIVER     56
     10                COMMON /INITSW/ LISTSF,POINTS,NOOFTP                               DRIVER     57
                       COMMON/ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                           DRIVER     58
                       COMMON/PTPOS7/ NOFPTS                                              DRIVER     59
                       COMMON/SETAB/ SURFTB(10,11)                                        DRIVER     60
                       COMMON /LASTPOS/ CHRPOS                                            DRIVER     61
     15                COMMON /PLTSZ/ VFACTR                                              DRIVER     62
                       COMMON /TAPE/ TAPENO,UNITNO                                        DRIVER     63
                       COMMON /INPTRC/ V(100),NOOFWD,NOOFRC,ROFMT(24)                     DRIVER     64
                       COMMON /COMPCOD/WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50  DRIVER     65
                       COMMON /DMBLK/IBLK(4235)                                           DRIVER     66
     20                DIMENSION LIST(6)                                                  DRIVER     67
                       DATA LIST/0,0,1,99,0,-1/                                           DRIVER     68
                       DATA TAPENO,UNITNO /1,1/                                           DRIVER     69
                       DATA NOFPTS /31/                                                   MODA        1
                 C                                                                        DRIVER     71
     25          C     INITIALIZE SYSTEM ERROR PROCESSOR FOR ERROR 78                     DRIVER     72
                 C       SEE APPENDIX G OF FORTRAN MANUAL                                 DRIVER     73
                 C                                                                        DRIVER     74
                       LIST(5) = LOCF(ERROR78)                                            DRIVER     75
                       CALL SYSTEMC( 78,LIST)                                             DRIVER     76
     30          C                                                                        DRIVER     77
                 C     INITIALIZE DATA HANDLER FILE AND HEAD LIST POINTER                 DRIVER     78
                 C                                                                        DRIVER     79
                       CALL DMTRL                                                         DRIVER     80
                      1CALL DMINIT(4HDATA,IBLK,4235,6)                                    DRIVER     81
     35          C                                                                        DRIVER     82
                 C        GET FIRST TWO WORD READ FOR TABLE POINTER STORAGE                DRIVER     83
                       CALL DMGT3D(2,IKNOW)                                               DRIVER     84
                       IHEADR=0                                                           DRIVER     85
                       UNITNO=7                                                           DRIVER     86
     40          C                                                                        DRIVER     87
                 C     SYSTEM INITIALIZATION                                              DRIVER     88
                       CALL OVERLAY(6HPLOTSL,1,0)                                         DRIVER     89
                       IF(LISTSF) WRITE(6,1000)                                           DRIVER     90
                 C                                                                        DRIVER     91
     45          C     TAPE INITIALIZATION                                                DRIVER     92
                       CALL OVERLAY(6HPLOTSL,2,0)                                         DRIVER     93
                 C                                                                        DRIVER     94
                 C         PLOT SPECIFICATION                                             DRIVER     95
                       CALL OVERLAY(6HPLOTSL,3,0)                                         DRIVER     96
     50          C                                                                        DRIVER     97
                 C         POINT COLLECTION                                               DRIVER     98
                       CALL OVERLAY(6HPLOTSL,4,0)                                         DRIVER     99
                       REWIND UNITNO                                                      DRIVER    100
                  100  CONTINUE                                                           DRIVER    101
     55          C                                                                        DRIVER    102
                 C         PRINTING OVERLAY                                               DRIVER    103
                       CALL OVERLAY(6HPLOTSL,5,0)                                         DRIVER    104
```

```
      C     PLOT GENERATION                                          DRIVER   105
      C                                                              DRIVER   106
   60         CALL OVERLAY(6HPLOTSL,6,0)                            DRIVER   107
              CALL DMFLSH                                           DRIVER   108
              STOP                                                  DRIVER   109
         1000 FORMAT(1H1,20X,19HTAPE INITIALIZATION)                 DRIVER   110
              END                                                   DRIVER   111
```

```
      SUBROUTINE SEARCH(TABLE,KEY,NAME)                                  DRIVER  112
C                                                                        DRIVER  113
C     SUBROUTINE PERFORMS BINARY SEARCH ON ORDERED TABLE STORED IN       DRIVER  114
C       BEADS                                                            DRIVER  115
C           TABLE IS POINTER TO HEADER BEAD OF TABLE                     DRIVER  116
C           KEY IS SEARCH BEAD                                           DRIVER  117
C           NAME IS SEARCH NAME ON ENTRY AND IS RETURNED AS NAME         DRIVER  118
C             SUBSCRIPT IF FOUND OR ZERO IF NOT                          DRIVER  119
C                                                                        DRIVER  120
      INTEGER TABLE,KEY                                                  DRIVER  121
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50          DRIVER  122
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 DRIVER  123
C                                                                        DRIVER  124
      CALL DMGET(WHOL1,TABLE,ISIZ)                                       DRIVER  125
      CALL DMGET(WHOL2+KEY,TABLE,KEYPTR)                                 DRIVER  126
      LOW=0                                                              DRIVER  127
      IHIGH=ISIZ+1                                                       DRIVER  128
      I=(IHIGH)/2                                                        DRIVER  129
    1 CONTINUE                                                           DRIVER  130
      CALL DMGET(WHOL0+I,KEYPTR,ITEMP)                                   DRIVER  131
      IF(NAME - ITEMP) 15,10,5                                           DRIVER  132
    5 LOW=I                                                              DRIVER  133
      I=(I+IHIGH)/2                                                      DRIVER  134
      IF(I-LOW) 1,50,1                                                   DRIVER  135
   15 IHIGH=I                                                            DRIVER  136
      I=(I+LOW+1)/2                                                      DRIVER  137
      IF(I-IHIGH) 1,50,1                                                 DRIVER  138
   10 CONTINUE                                                           DRIVER  139
      NAME=I                                                             DRIVER  140
      RETURN                                                             DRIVER  141
   50 CONTINUE                                                           DRIVER  142
      NAME=0                                                             DRIVER  143
      RETURN                                                             DRIVER  144
      END                                                                DRIVER  145
```

```
      SUBROUTINE SORTAB(TABLE,KEY)                                              DRIVER  146
C                                                                               DRIVER  147
C     ROUTINE SORTS TABLE STORED IN BEADS USING SHELL SORT                      DRIVER  148
C                                                                               DRIVER  149
C     TABLE LAYOUT                                                              DRIVER  150
C                                                                               DRIVER  151
C          TABLE------* TABLE SIZE  *                                          DRIVER  152
C                     * NO OF COL   *                                          DRIVER  153
C                     * PTR TO COL1 *                                          DRIVER  154
C                     * PTR TO COL2 *                                          DRIVER  155
C                     *    ...      *                                          DRIVER  156
C                     * PTR TO COLN *                                          DRIVER  157
C                                                                               DRIVER  158
      INTEGER TABLO,TABHI,TEMP1,TEMP2,XCHPTR,TABLE                              DRIVER  159
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50                 DRIVER  160
      INTEGER FWDPOS,FWDSTP,REVPOS,REVSTP,SEPDIS                                DRIVER  161
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50        DRIVER  162
      EQUIVALENCE(FWDSTP,REVSTP)                                                DRIVER  163
C                                                                               DRIVER  164
      CALL DMGET(WHOL1,TABLE,NOOFEL)                                            DRIVER  165
      CALL DMGET(WHOL2,TABLE,NOOFCL)                                            DRIVER  166
      CALL DMGET(WHOL2+KEY,TABLE,KEYPTR)                                        DRIVER  167
C                                                                               DRIVER  168
      IF(NOOFEL .LE. 1) RETURN                                                  DRIVER  169
      SEPDIS=(NOOFEL+1)/2                                                       DRIVER  170
   10 CONTINUE                                                                  DRIVER  171
      FWDPOS=1                                                                  DRIVER  172
   20 CONTINUE                                                                  DRIVER  173
      FWDSTP=FWDPOS+SEPDIS                                                      DRIVER  174
      IF(FWDSTP .GT. NOOFEL) GO TO 50                                           DRIVER  175
      CALL DMGET(WHOL0+FWDPOS,KEYPTR,TABLO)                                     DRIVER  176
      CALL DMGET(WHOL0+FWDSTP,KEYPTR,TABHI)                                     DRIVER  177
      IF(TABLO .GT. TABHI) GO TO 30                                             DRIVER  178
   25 CONTINUE                                                                  DRIVER  179
      FWDPOS=FWDPOS+1                                                           DRIVER  180
      GO TO 20                                                                  DRIVER  181
   30 CONTINUE                                                                  DRIVER  182
      I=REVPOS=FWDPOS                                                           DRIVER  183
      K=FWDSTP                                                                  DRIVER  184
   40 CONTINUE                                                                  DRIVER  185
      DO 45 L=1,NOOFCL                                                          DRIVER  186
           CALL DMGET(WHOL2+L,TABLE,XCHPTR)                                     DRIVER  187
           CALL DMGET(WHOL0+I,XCHPTR,TEMP1)                                     DRIVER  188
           CALL DMGET(WHOL0+K,XCHPTR,TEMP2)                                     DRIVER  189
           CALL DMSET(WHOL0+I,XCHPTR,TEMP2)                                     DRIVER  190
           CALL DMSET(WHOL0+K,XCHPTR,TEMP1)                                     DRIVER  191
   45 CONTINUE                                                                  DRIVER  192
      REVSTP=REVPOS                                                             DRIVER  193
      REVPOS=REVPOS-SEPDIS                                                      DRIVER  194
      IF(REVPOS .LE. 0) GO TO 25                                                DRIVER  195
      CALL DMGET(WHOL0+REVSTP,KEYPTR,TABHI)                                     DRIVER  196
      CALL DMGET(WHOL0+REVPOS,KEYPTR,TABLO)                                     DRIVER  197
      IF(TABHI .GE. TABLO) GO TO 25                                             DRIVER  198
      I=REVPOS                                                                  DRIVER  199
      K=REVSTP                                                                  DRIVER  200
      GO TO 40                                                                  DRIVER  201
   50 CONTINUE                                                                  DRIVER  202
```

```
      SEPDIS=SEPDIS/2                                          DRIVER   203
      IF(SEPDIS .LE. 0) RETURN                                 DRIVER   204
60    GO TO 10                                                 DRIVER   205
      END                                                      DRIVER   206
```

```
      INTEGER FUNCTION ZTRUNC(IARG,IPOS)                                DRIVER  207
C                                                                       DRIVER  208
C        IARG  -  INPUT WORD                                            DRIVER  209
C        IPOS  -  POSITION OF CHARACTER IN INPUT WORD TO BE PRESERVED   DRIVER  210
C                                                                       DRIVER  211
C        THIS FUNCTION RETURNS THE IPOS CHARACTER LEFT JUSTIFIED AND    DRIVER  212
C        ZERO FILLED                                                    DRIVER  213
      DATA IMASK/77000000000000000000B/                                 DRIVER  214
      INARG = IARG                                                      DRIVER  215
      N = (IPOS - 1) * 6                                                DRIVER  216
      ZTRUNC = SHIFT(INARG,N)                                           DRIVER  217
      ZTRUNC = ZTRUNC .AND. IMASK                                       DRIVER  218
      RETURN                                                            DRIVER  219
      END                                                               DRIVER  220
```

```
      1              BLOCK DATA INITL                                                    DRIVER    221
                     INTEGER FLETAB,FUNCTB,FUNTAB                                        DRIVER    222
                     INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50           DRIVER    223
                     COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50  DRIVER    224
      5              COMMON/DFLTVL/ TOL(145),INITAX(145,4),FUNTAB(10,3)                  DRIVER    225
                     COMMON /ITABLE/ FLETAB,FUNCTB,XSCAL,YSCAL                           DRIVER    226
                C                                                                        DRIVER    227
                C                                                                        DRIVER    228
                C    CODE DIGIT VALUES FOR ELEMENT TABLE SEE SUB ELEMST                  DRIVER    229
     10         C                                                                        DRIVER    230
                C    REGULAR ELEMENTS ARE ASSIGNED CODE NUMBERS FROM 1 TO 100            DRIVER    231
                C    SPECIAL FUNCTIONS ARE ASSIGNED CODE NUMBERS FROM 501 TO 599         DRIVER    232
                C    SURFACES ARE ASSIGNED CODE NUMBERS FROM 601 TO 699                  DRIVER    233
                C                                                                        DRIVER    234
     15         COMMON/IOUNIT/IN,IO,IN1,IO1,IN2,IO2,IN3,IO3                              DRIVER    235
                COMMON/PLTSZ/VFACTR                                                      DRIVER    236
                COMMON/INITSW/LISTSF,POINTS,NOOFTP                                       DRIVER    237
                COMMON/INPTRC/V(100),NOOFWD,NOOFRC,RDFMT(24)                             DRIVER    238
                COMMON/HEADPTR/IFPG                                                      DRIVER    239
     20         COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONCRD                               DRIVER    240
                COMMON/CCODES/IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6            DRIVER    241
                COMMON/ERROR/CHECK                                                       DRIVER    242
                C                                                                        PG          2
                C       PLOT GROUP BEAD                                                  PG          3
     25         COMMON /PG/ IPGNXT,IPGLST,IPGTT,IPGAB,IPGCT,IPGNCT,IPGLAB,               PG          4
               X            IPGNLB,IPGLB                                                PG          5
                C                                                                        AB          2
                C       AXIS BEAD                                                        AB          3
                COMMON /AB/ IABNXT,IABLST,IABT,IABPM,IABXN,IABNY                         AB          4
     30         C                                                                        PM          2
                C       PARAMETER - MULTIPLIER BEAD                                      PM          3
                COMMON /PM/ IPMNXT,IPMLST,IPMPL,IPMML,IPMPB,IPMPT,IPMMT                  PM          4
                C                                                                        PB          2
                C       PLOT BEAD                                                        PB          3
     35         COMMON /PB/ IPBNXT,IPBNT,IPBLST,IPBLT,IPBPG,IPBTT,IPBAB,IPBAS,           PB          4
               X            IPBPN,IPBLB,IPBCB,IPBNM                                     PB          5
                C                                                                        CB          2
                C       CURVE BEAD                                                       CB          3
                COMMON /CB/ ICBNXT,ICBLST,ICBPN,ICBPV,ICBNPT,ICBSYM,ICBRB,ICBXB,         CB          4
     40        X            ICBYB                                                       CB          5
                DATA POINTS,NOOFTP,LISTSF/.FALSE.,1,.FALSE./                             DRIVER    243
                DATA VFACTR/1.0/                                                         DRIVER    244
                DATA NOOFWD/20/                                                          DRIVER    245
                DATA (RDFMT(I),I=1,24)/10H(F4.0,F5.0, 10H,F3.0,F11.,                     DRIVER    246
     45        X 10H0,F6.3,F7., 10H2,F9.0,F6., 10H2,F6.2,F7., 10H4,F8.5,F7.,            DRIVER    247
               X 10H4,F9.5,F8., 10H5,F8.5,F7., 10H4,F7.4,F5., 10H2,F7.4,F3., 2H0),      DRIVER    248
               X 13*(1H )/                                                               DRIVER    249
                DATA IFPG,LPG,LAXIS,LPM,IESWT/4*(0),1/                                   DRIVER    250
                DATA IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4/0100000000018,0100000000028,        DRIVER    251
     50        X      0100000000039,0100000000048/                                      DRIVER    252
                DATA IWHOLE5,IWHOLE6/0100000000058,0100000000068/                        DRIVER    253
                DATA LCONCRD/0/                                                          DRIVER    254
                DATA IN,IN1,IN2,IN3/4*5/                                                 DRIVER    255
                DATA IO,IO1,IO2,IO3/4*6/                                                 DRIVER    256
     55         C                                                                        DRIVER    257
                C                                                                        DRIVER    258
                C    VALUES FOR FUNCTION TABLE                                           DRIVER    259
```

```
      C     (POSITION 1=FUNCTION NAME 2=FUNCTION CODE 3=NO OF ARGS)                 DRIVER  260
      C     NO OF ARGS=999 MEANS NO LIMIT TO NUMBER                                 DRIVER  261
   60 C                                                                             DRIVER  262
      DATA (FUNTAB(1,I),I=1,3) /2HEQ,5,999/                                         DRIVER  263
      DATA (FUNTAB(2,I),I=1,3) /2HGE,3,1/                                           DRIVER  264
      DATA (FUNTAB(3,I),I=1,3) /2HGT,4,1/                                           DRIVER  265
      DATA (FUNTAB(4,I),I=1,3) /2HLE,1,1/                                           DRIVER  266
   65 DATA (FUNTAB(5,I),I=1,3) /2HLT,2,1/                                           DRIVER  267
      DATA (FUNTAB(6,I),I=1,3) /2HNE,6,999/                                         DRIVER  268
      DATA (FUNTAB(7,I),I=1,3) /5HRANGE,7,2/                                        DRIVER  269
      C                                                                             DRIVER  270
      C     DEFAULT VALUES FOR TOLERANCE                                            DRIVER  271
   70 C     VALUES SHOULD BE IN ORDER OF INCREASING CODE DIGIT                      DRIVER  272
      C     1-100                                                                   DRIVER  273
      C     501-535                                                                 DRIVER  274
      C     601-610                                                                 DRIVER  275
      C                                                                             DRIVER  276
   75 DATA (TOL(I),I=1,145) /4*0.,.05,2.,.1E6,2*.4,.001,.0001,.001,                 DRIVER  277
     X2*.0001,2*.001,.0001,.01,.001,.01,80*0.,.001,2*.0001,.001,                    DRIVER  278
     X2*.0001,.001,.0001,.01,.3,35*0./                                              DRIVER  279
      C                                                                             DRIVER  280
      C     INITIALIZATION VALUES FOR AXES SCALES                                   DRIVER  281
   80 C                                                                             DRIVER  282
      C     AXES ARRAY FORMAT                                                       DRIVER  283
      C     SUBSCRIPT 1 ELEMENT IN ORDER OF INCREASING CODE DIGIT                   DRIVER  284
      C     SUBSCRIPT 2 X MIN X INCR Y MIN Y INCR                                   DRIVER  285
      C                                                                             DRIVER  2
   85 DATA XSCAL,YSCAL /9.,6./                                                      DRIVER  2
      C                                                                             DRIVER  288
      C     V1 TO V100, CODE = 1-100                                                DRIVER  289
      C     C1 TO C15, CODE = 501-515                                               DRIVER  290
      C     FUN1 TO FUN20, CODE = 516-535                                           DRIVER  291
   90 C     SURF1 TO SURF10, CODE = 601-610                                         DRIVER  292
      C                                                                             DRIVER  293
      DATA (INITAX( 1,I),I=1,4) /0.,0.,0.,0./                                       DRIVER  294
      DATA (INITAX( 2,I),I=1,4) /0.,0.,0.,0./                                       DRIVER  295
      DATA (INITAX( 3,I),I=1,4) /0.,0.,0.,0./                                       DRIVER  296
   95 DATA (INITAX( 4,I),I=1,4) /0.,0.,0.,0./                                       DRIVER  297
      DATA (INITAX( 5,I),I=1,4) /0.,.4,0.,.4/                                       DRIVER  298
      DATA (INITAX( 6,I),I=1,4) /3.,100.,0.,200./                                   DRIVER  299
      DATA (INITAX( 7,I),I=1,4) /1.E+6,4.E+6,1.E+6,8.E+6/                           DRIVER  300
      DATA (INITAX( 8,I),I=1,4) /-9.,2.,-12.,4./                                    DRIVER  301
  100 DATA (INITAX( 9,I),I=1,4) /-9.,8.,-4.,4./                                    DRIVER  302
      DATA (INITAX( 10,I),I=1,4) /-.3,.2,-.4,.2/                                    DRIVER  303
      DATA (INITAX( 11,I),I=1,4) /-.04,.02,-.06,.04/                                DRIVER  304
      DATA (INITAX( 12,I),I=1,4) /-.04,.01,-.06,.02/                                DRIVER  305
      DATA (INITAX( 13,I),I=1,4) /+.04,-.01,-.06,.02/                               DRIVER  306
  105 DATA (INITAX( 14,I),I=1,4) /-.04,.01,-.03,.01/                                DRIVER  307
      DATA (INITAX( 15,I),I=1,4) /-.04,.01,-.03,.01/                                DRIVER  308
      DATA (INITAX( 16,I),I=1,4) /-.8,.2,-.4,.2/                                    DRIVER  309
      DATA (INITAX( 17,I),I=1,4) /0.,.2,0.,.2/                                      DRIVER  310
      DATA (INITAX( 18,I),I=1,4) /-9.,2.,-4.,2./                                    DRIVER  311
  110 DATA (INITAX( 19,I),I=1,4) /-9.,2.,-9.,4./                                    DRIVER  312
      DATA (INITAX( 20,I),I=1,4) /-9.,2.,-9.,4./                                    DRIVER  313
      C                                                                             DRIVER  314
      C     VARIABLES V21 THRU V100 HAVE NO DEFAULT SCALES ASSIGNED                 DRIVER  315
      DATA ((INITAX(I,J),J=1,4),I=21,100) /320*0./                                  DRIVER  316
```

```
      DATA (INITAX(101,I),I=1,4) /-.04,.01,-.06,.02/                    DRIVER    317
      DATA (INITAX(102,I),I=1,4) /-.04,.02,-.06,.04/                    DRIVER    318
      DATA (INITAX(103,I),I=1,4) /-.04,.01,-.03,.01/                    DRIVER    319
      DATA (INITAX(104,I),I=1,4) /-.04,.01,-.03,.01/                    DRIVER    320
      DATA (INITAX(105,I),I=1,4) /-.04,.01,-.03,.01/                    DRIVER    321
      DATA (INITAX(106,I),I=1,4) /0.,.05,0.,.05/                        DRIVER    322
      DATA (INITAX(107,I),I=1,4) /-.8,.2,-.4,.2/                        DRIVER    323
      DATA (INITAX(108,I),I=1,4) /0.,.2,0.,.2/                          DRIVER    324
      DATA (INITAX(109,I),I=1,4) /-8.,2.,-4.,2./                        DRIVER    325
      DATA (INITAX(110,I),I=1,4) /0.,5.,0.,10./                         DRIVER    326
C                                                                       DRIVER    327
C     VARIABLES C11 THRU FUN20 HAVE NO DEFAULT SCALES ASSIGNED          DRIVER    328
      DATA ((INITAX(I,J),J=1,4),I=111,135) /100*0./                     DRIVER    329
      DATA ((INITAX(I,J),J=1,4),I=136,145) /10*(-80.,20.,-60.,20.)/     DRIVER    330
C                                                                       DRIVER    331
C     COMPONENT CODE VALUES                                             DRIVER    332
C     SEE GRAPHICS MANUAL P 7-48                                        DRIVER    333
C                                                                       DRIVER    334
      DATA WHOL0 /0100000000038/                                        DRIVER    335
      DATA WHOL1 /0100000000018/                                        DRIVER    336
      DATA WHOL2 /0100000000028/                                        DRIVER    337
      DATA WHOL3 /0100000000038/                                        DRIVER    338
      DATA WHOL4 /0100000000048/                                        DRIVER    339
      DATA WHOL5 /0100000000058/                                        DRIVER    340
      DATA ARRY20 /0700240000018/                                       DRIVER    341
      DATA ARRY50 /0700620000018/                                       DRIVER    342
      DATA IPGNXT,IPGLST,IPGTT,IPGAB,IPGCT,IPGNCT,IPGLAB,IPGNLB,IPGLB / MODA        7
     X     1000000000018,1000000000028,1000000000038,1000000000048,     MODA        8
     X     1000000000058,1000000000068,1000000000078,1000000000108,     MODA        9
     X     1000000000118 /                                              MODA       10
      DATA IABNXT,IABLST,IABT,IABPM,IABXN,IABNY /                       MODA       11
     X     1000000000018,1000000000028,1000000000038,1000000000048,     MODA       12
     Y     1000000000058,1000000000068 /                                MODA       13
      DATA IPMNXT,IPMLST,IPMPL,IPMML,IPMPB,IPMPT,IPMMT /                MODA       14
     X     1000000000018,1000000000028,1000000000038,1000000000048,     MODA       15
     X     1000000000058,1000000000068,1000000000078 /                  MODA       16
      DATA IPBNXT,IPBNT,IPBLST,IPBLT,IPBPG,IPBTT,IPBAB,IPBAS,IPBPM,     MODA       17
     X     IPBLB,IPBCB,IPBNM /                                          MODA       18
     X     1000000000018,1000000000028,1000000000038,1000000000048,     MODA       19
     X     1000000000058,1000000000068,1000000000078,1000000000108,     MODA       20
     X     1000000000118,1000000000128,1000000000138,1000000000148 /    MODA       21
      DATA ICBNXT,ICBLST,ICBPN,ICBPV,ICBNPT,ICBSYM,ICBRB,ICBXB,ICBYB /  MODA       22
     X     1000000000018,1000000000028,1000000000038,1000000000048,     MODA       23
     X     1000000000058,1000000000068,1000000000078,1000000000108,     MODA       24
     X     1000000000118 /                                              MODA       25
      END                                                               DRIVER    343
```

```
      OVERLAY(1,0)                                                      INITL    2
      SUBROUTINE AXES(INPUT)                                            INITL    3
      INTEGER ZTRUNC,ARG,CHRPOS                                         INITL    4
      DIMENSION INPUT(80),NAMARG(2)                                     INITL    5
      COMMON /LASTPOS/ CHRPOS                                           INITL    6
      DATA (NAMARG(I),I=1,2) /1LR,1LS/                                  INITL    7
C                                                                       INITL    8
C     SUBROUTINE HANDLES DECODING OF ARGUMENTS FOR INITIALIZING AXES    INITL    9
C     ARGUMENT IS  (RESTORE,SAME)                                       INITL   10
C                                                                       INITL   11
C     DECODE ARGUMENT NAME OR VALUE                                     INITL   12
C                                                                       INITL   13
  100 CONTINUE                                                          INITL   14
      CALL SCAN(CHRPOS,INPUT,ARG)                                       INITL   15
      IF(ARG .EQ. 0) GO TO 420                                          INITL   16
C                                                                       INITL   17
C     SELECT ARGUMENT                                                   INITL   18
C       I  =1,RESTORE  =2,SAME                                          INITL   19
C                                                                       INITL   20
  200 CONTINUE                                                          INITL   21
      ARG=ZTRUNC(ARG,1)                                                 INITL   22
      DO 210 I=1,2                                                      INITL   23
      IF(ARG .EQ. NAMARG(I)) GO TO (410,420),I                          INITL   24
  210 CONTINUE                                                          INITL   25
      GO TO 511                                                         INITL   26
C                                                                       INITL   27
C     PERFORM INITIALIZATION OPTION                                     INITL   28
C                                                                       INITL   29
  400 CONTINUE                                                          INITL   30
C                                                                       INITL   31
C     RESTORE ELEMENT TABLE TO DEFAULT VALUES                           INITL   32
C                                                                       INITL   33
  410 CONTINUE                                                          INITL   34
      CALL AXESST                                                       INITL   35
C                                                                       INITL   36
C     USE SAME AXES TABLE  TRANSFER TO SUBROUTINE WHICH INITIALIZES     INITL   37
C     AXES                                                              INITL   38
C                                                                       INITL   39
  420 CONTINUE                                                          INITL   40
      CALL INITAX                                                       INITL   41
      RETURN                                                            INITL   42
C                                                                       INITL   43
C     ERROR IN 3RD ARGUMENT VALUE,  SAME ASSUMED                        INITL   44
C                                                                       INITL   45
  511 CALL ERRORM(37H(32H ARG3=RESTORE,SAME  SAME ASSUMED),1,4HAXES)    INITL   46
      GO TO 420                                                         INITL   47
      END                                                               INITL   48
```

```
      SUBROUTINE AXESST                                                  INITL  49
C                                                                        INITL  50
C     ROUTINE SETS AXIS SCALE FACTORS TO DEFAULT VALUES                  INITL  51
C                                                                        INITL  52
      INTEGER ELETAB,FUNCTB,COLPTR,FUNTAB                                INITL  53
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50          INITL  54
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 INITL  55
      COMMON /DFLTVL/ TOL(145),INITAX(145,4),FUNTAB(10,3)                INITL  56
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                          INITL  57
C                                                                        INITL  58
      CALL SORTAB(ELETAB,2)                                              INITL  59
      CALL DMGET(WHOL1,ELETAB,NOOFAX)                                    INITL  60
      DO 20 I=1,NOOFAX                                                   INITL  61
           DO 10 J=1,4                                                   INITL  62
                CALL DMGET(WHOL5+J,ELETAB,COLPTR)                        INITL  63
                CALL DMSET(WHOL0+I,COLPTR,INITAX(I,J))                   INITL  64
   10           CONTINUE                                                 INITL  65
   20      CONTINUE                                                      INITL  66
      CALL SORTAB(ELETAB,1)                                              INITL  67
      RETURN                                                             INITL  68
      END                                                                INITL  69
```

```
      SUBROUTINE DEFELE                                                 INITL    70
C                                                                       INITL    71
C     ROUTINE GETS BEADS FOR ELEMENT TABLE                              INITL    72
C       FORMAT IS                                                       INITL    73
C                                                                       INITL    74
C          ELETAB-------* NO OF ELEMENTS        *                      INITL    75
C                        * NO OF COLUMNS        *                      INITL    76
C                        * PTR TO NAME BD       *                      INITL    77
C                        * PTR TO CODE DIGIT BD *                      INITL    78
C                        * PTR TO TOL BD        *                      INITL    79
C                        * PTR TO X MIN BD      *                      INITL    80
C                        * PTR TO X INCR BD     *                      INITL    81
C                        * PTR TO Y MIN BD      *                      INITL    82
C                        * PTR TO Y INCR BD     *                      INITL    83
      INTEGER ELETAB,PTR,FUNCTB                                         INITL    84
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50         INITL    85
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 INITL   86
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                         INITL    87
      DATA NOFEL,NOFCOL /145,7/                                         INITL    88
C                                                                       INITL    89
C     GET HEADER BEAD AND BEADS TO CONTAIN NAME, CODE DIGIT,TOL,X MIN,  INITL    90
C       X INCR, Y MIN, YINCR                                            INITL    91
C                                                                       INITL    92
      CALL DMGTBD(9,ELETAB)                                             INITL    93
      CALL DMSET(WHOL1,ELETAB,NOFEL)                                    INITL    94
      CALL DMSET(WHOL2,ELETAB,NOFCOL)                                   INITL    95
      DO 10 I=1,NOFCOL                                                  INITL    96
           CALL DMGTBD(NOFEL,PTR)                                       INITL    97
           CALL DMSET(WHOL2+I,ELETAB,PTR)                               INITL    98
   10      CONTINUE                                                     INITL    99
      CALL ELEMST                                                       INITL   100
      RETURN                                                            INITL   101
      END                                                               INITL   102
```

```
      SUBROUTINE DEFFUN                                                 INITL  103
C                                                                       INITL  104
C     ROUTINE STORES FUNCTION NAMES,CODE DIGITS IN TABLE HEADS          INITL  105
C                                                                       INITL  106
      INTEGER ELETAB,FUNCTB,FUNTAB,COLPTR(3)                            INITL  107
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50         INITL  108
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 INITL  109
      COMMON /DFLTVL/ TOL(145),INITAX(145,4),FUNTAB(10,3)               INITL  110
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                         INITL  111
      DATA NOFUNT,NOFCOL /7,3/                                          INITL  112
C                                                                       INITL  113
C     GET HEADER HEAD AND HEADS TO CONTAIN NAME,CODE DIGIT,AND NO OF ARG INITL  114
C                                                                       INITL  115
      CALL DMGTBD(5,FUNCTB)                                             INITL  116
C                                                                       INITL  117
C     COPY INFORMATION INTO TABLE HEADS                                 INITL  118
C                                                                       INITL  119
      DO 10 I=1,NOFCOL                                                  INITL  120
         CALL DMGTBD(NOFUNT,COLPTR(I))                                  INITL  121
         CALL DMSET(WHOL2+I,FUNCTB,COLPTR(I))                           INITL  122
   10    CONTINUE                                                       INITL  123
      CALL DMSET(WHOL1,FUNCTB,NOFUNT)                                   INITL  124
      CALL DMSET(WHOL2,FUNCTB,NOFCOL)                                   INITL  125
      DO 20 I=1,NOFUNT                                                  INITL  126
         DO 15 J=1,NOFCOL                                               INITL  127
            CALL DMSET(WHOL0+I,COLPTR(J),FUNTAB(I,J))                   INITL  128
   15       CONTINUE                                                    INITL  129
   20    CONTINUE                                                       INITL  130
C                                                                       INITL  131
C     SORT TABLE AFTER INITIALIZATION                                   INITL  132
C                                                                       INITL  133
      CALL SORTAB(FUNCT3,1)                                             INITL  134
      RETURN                                                            INITL  135
      END                                                               INITL  136
```

```
      SUBROUTINE ELMENT(INPUT)                                          INITL  137
      INTEGER ZTRUNC,ARG,CHRPOS                                         INITL  138
      DIMENSION INPUT(80),NAMARG(2)                                     INITL  139
      COMMON /LASTPOS/ CHRPOS                                           INITL  140
      DATA (NAMARG(I),I=1,2) /1LR,1LS/                                  INITL  141
C                                                                       INITL  142
C     SUBROUTINE HANDLES DECODING OF ARGUMENTS FOR INITIALIZING ELEMENTS INITL  143
C     ARGUMENT IS (RESTORE,SAME)                                        INITL  144
C                                                                       INITL  145
C     DETERMINE ARGUMENT NAME OR VALUE                                  INITL  146
C                                                                       INITL  147
  100 CONTINUE                                                          INITL  148
      CALL SCAN(CHRPOS,INPUT,ARG)                                       INITL  149
      IF(ARG .EQ. 0) GO TO 420                                          INITL  150
C                                                                       INITL  151
C     SELECT ARGUMENT                                                   INITL  152
C       I  =1,RESTORE  =2,SAME                                          INITL  153
C                                                                       INITL  154
  200 CONTINUE                                                          INITL  155
      ARG=ZTRUNC(ARG,1)                                                 INITL  156
      DO 210 I=1,2                                                      INITL  157
      IF(ARG .EQ. NAMARG(I)) GO TO (410,420),I                          INITL  158
  210 CONTINUE                                                          INITL  159
      GO TO 511                                                         INITL  160
C                                                                       INITL  161
C     PROCESSING COMPLETED PERFORM INITIALIZATION OPTION                INITL  162
C                                                                       INITL  163
  400 CONTINUE                                                          INITL  164
C                                                                       INITL  165
C     RESTORE ELEMENT TABLE TO DEFAULT VALUES                           INITL  166
C                                                                       INITL  167
  410 CONTINUE                                                          INITL  168
      CALL ELEMST                                                       INITL  169
C                                                                       INITL  170
C     USE SAME ELEMENT TABLE  TRANSFER TO SUBROUTINE WHICH INITIALIZES  INITL  171
C     ELEMENTS                                                          INITL  172
C                                                                       INITL  173
  420 CONTINUE                                                          INITL  174
      CALL INITEL                                                       INITL  175
      RETURN                                                            INITL  176
C                                                                       INITL  177
C     ERROR IN 3RD ARGUMENT VALUE,  SAME ASSUMED                        INITL  178
C                                                                       INITL  179
  511 CALL ERRORM(37H(32H ARG3=RESTORE,SAME  SAME ASSUMED),1,6HELMENT)  INITL  180
      GO TO 420                                                         INITL  181
      END                                                               INITL  182
```

```
      SUBROUTINE ELEMST                                                 INITL  183
C                                                                       INITL  184
C     ROUTINE INITIALIZES ELEMENT TABLE TO DEFAULT VALUES               INITL  185
C                                                                       INITL  186
      INTEGER AXSPTR(4),FUNCTB,TOLPTR,TABPOS                            INITL  187
      INTEGER ELETAB,FUNTAB,DIGPTR,FORMT(8)                             INITL  188
      INTEGER WHOLC,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50         INITL  189
C                                                                       INITL  190
      COMMON /COMPCOD/ WHOLC,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 INITL  191
      COMMON /DFLTVL/ TOL(145),INITAX(145,4),FUNTAB(10,3)               INITL  192
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                         INITL  193
C                                                                       INITL  194
      DATA NOFVAR,NOFSUR,NOFFUN,NOSPVR/100 ,10,20,15/                   INITL  195
      DATA FORMT(1) /18H(1HV,A3,6H       )/                           INITL  196
      DATA FORMT(3) /19H(4HSURF,A2,4H    )/                           INITL  197
      DATA FORMT(5) /18H(3HFUN,A2,6H     )/                           INITL  198
      DATA FORMT(7) /18H(1HC,A2,7H       )/                           INITL  199
C                                                                       INITL  200
      TABPOS=1                                                          INITL  201
      CALL DMGET(WHOL3,ELETAB,NAMPTR)                                   INITL  202
      CALL DMGET(WHOL4,ELETAB,DIGPTR)                                   INITL  203
C                                                                       INITL  204
C     STORE DEFAULT ELEMENT NAMES AND CODE DIGITS                       INITL  205
C                                                                       INITL  206
      CALL STRTAB(TABPOS,1,NOFVAR,NAMPTR,FORMT(1),DIGPTR,0)             INITL  207
C                                                                       INITL  208
C     STORE DEFAULT SPECIAL ELEMENT NAMES AND CODE DIGITS               INITL  209
C                                                                       INITL  210
      CALL STRTAB(TABPOS,1,NOSPVR,NAMPTR,FORMT(7),DIGPTR,500)           INITL  211
C                                                                       INITL  212
C     STORE DEFAULT FUNCTION NAMES AND CODE DIGITS                      INITL  213
C                                                                       INITL  214
      CALL STRTAB(TABPOS,1,NOFFUN,NAMPTR,FORMT(5),DIGPTR,515)           INITL  215
C                                                                       INITL  216
C     STORE DEFAULT SURFACE NAMES AND CODE DIGITS                       INITL  217
C                                                                       INITL  218
      CALL STRTAB(TABPOS,1,NOFSUR,NAMPTR,FORMT(3),DIGPTR,600)           INITL  219
C                                                                       INITL  220
C     ELEMENTS ARE IN ORDER OF CODE DIGIT FOR SETTING DEFAULT ENTRIES   INITL  221
C                                                                       INITL  222
C     STORE DEFAULT TOLERANCE VALUES IN TABLE                           INITL  223
C                                                                       INITL  224
      CALL DMGET(WHOL1,ELETAB,NOFWD)                                    INITL  225
      CALL DMGET(WHOL5,ELETAB,TOLPTR)                                   INITL  226
      DO 10 I=1,NOFWD                                                   INITL  227
         CALL DMSET(WHOLC+I,TOLPTR,TOL(I))                              INITL  228
   10    CONTINUE                                                       INITL  229
C                                                                       INITL  230
C     STORE DEFAULT AXES SCALE SETTINGS IN TABLE                        INITL  231
C                                                                       INITL  232
      DO 20 I=1,4                                                       INITL  233
         CALL DMGET(WHOL5+I,ELETAB,AXSPTR(I))                           INITL  234
   20    CONTINUE                                                       INITL  235
      DO 40 I=1,NOFWD                                                   INITL  236
         DO 30 J=1,4                                                    INITL  237
         CALL DMSET(WHOLC+I,AXSPTR(J),INITAX(I,J))                      INITL  238
   30       CONTINUE                                                    INITL  239
```

```
                40      CONTINUE                                              INITL  240
                C                                                             INITL  241
60              C       SORT ELEMENT TABLE BY NAME                            INITL  242
                C                                                             INITL  243
                        CALL SORTAB(ELETAB,1)                                 INITL  244
                        RETURN                                                INITL  245
                        END                                                   INITL  246
```

```
      SUBROUTINE INITAX                                                 INITL  247
      LOGICAL RDCHK                                                     INITL  248
      INTEGER CHRPOS,ZTRUNC,ARG,ARGSWT,FINISH                           INITL  249
      INTEGER COLPTR,ELETAB,FUNCTB                                      INITL  250
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50         INITL  251
      COMMON/COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 INITL  252
      COMMON/ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                          INITL  253
      COMMON /ERROR/ RDCHK                                              INITL  254
      COMMON /LASTPOS/ CHRPOS                                           INITL  255
      DIMENSION TABLE(5),ITABLE(5),INPUT(80)                            INITL  256
      DIMENSION NAMAG1(4),NAMAG2(4),NAMAG3(4)                           INITL  257
      EQUIVALENCE(TABLE,ITABLE)                                         INITL  258
      DATA (NAMAG1(I),I=1,4) /1LO,1LX,1LY,1L /                          INITL  259
      DATA (NAMAG2(I),I=1,4) /1LM,1LI,1LY,1LO/                          INITL  260
      DATA (NAMAG3(I),I=1,4) /1LX,1LM,1LI,1LO/                          INITL  261
      DATA ID,FINISH,NOTSET /6HINITAX,1LF,1H /                          INITL  262
C                                                                       INITL  263
C     SUBROUTINE INITIALIZES AXES                                       INITL  264
C     ARGUMENTS ARE OLD=NAME X INCR=NO MIN=NO Y MIN=NO INCR=NO          INITL  265
C                   FINISH                                              INITL  266
C                                                                       INITL  267
C     SET COUNTERS, ARRAYS, AND SWITCHES                                INITL  268
C                                                                       INITL  269
   10 CONTINUE                                                          INITL  270
      CHRPOS=1                                                          INITL  271
      INDEXO=NOTSET                                                     INITL  272
      ISWTCH=1                                                          INITL  273
      RDCHK=.FALSE.                                                     INITL  274
      ARGSWT=1                                                          INITL  275
      DO 20 I=1,5                                                       INITL  276
      ITABLE(I)=NOTSET                                                  INITL  277
   20 CONTINUE                                                          INITL  278
C                                                                       INITL  279
C     READ INITIALIZATION CARD                                          INITL  280
C                                                                       INITL  281
      CALL READCD(INPUT)                                                INITL  282
C                                                                       INITL  283
C     DECODE ARGUMENT NAME OR VALUE                                     INITL  284
C                                                                       INITL  285
  100 CONTINUE                                                          INITL  286
      CALL SCAN(CHRPOS,INPUT,ARG)                                       INITL  287
      IF(ARG .EQ. 0) GO TO 400                                          INITL  288
      GO TO (200,300),ISWTCH                                            INITL  289
C                                                                       INITL  290
C     SELECT ARGUMENT                                                   INITL  291
C                                                                       INITL  292
  200 CONTINUE                                                          INITL  293
      ARG=ZTRUNC(ARG,1)                                                 INITL  294
      DO 210 I=1,4                                                      INITL  295
      GO TO (202,204,206),ARGSWT                                        INITL  296
  202 IF(ARG .EQ. NAMAG1(I)) GO TO (230,233,236,512),I                  INITL  297
      GO TO 210                                                         INITL  298
  204 IF(ARG .EQ. NAMAG2(I)) GO TO (240,240,236,230),I                  INITL  299
      GO TO 210                                                         INITL  300
  206 IF(ARG .EQ. NAMAG3(I)) GO TO (233,240,240,230),I                  INITL  301
  210 CONTINUE                                                          INITL  302
      IF(ARG .EQ. FINISH) RETURN                                        INITL  303
```

```
                      GO TO 512                                                 INITL  304
                C                                                               INITL  305
     60         C     CHECK FOR PREVIOUS SETTING OF OLD NAME                    INITL  306
                C                                                               INITL  307
                  230 CONTINUE                                                  INITL  308
                      NAMVAL=1                                                  INITL  309
                      IF(ITABLE(NAMVAL) .NE. NOTSET) GO TO 511                  INITL  310
     65               ISWTCH=2                                                  INITL  311
                      ARGSWT=1                                                  INITL  312
                      GO TO 100                                                 INITL  313
                C                                                               INITL  314
                C     X DECODE                                                  INITL  315
     70         C                                                               INITL  316
                  233 CONTINUE                                                  INITL  317
                      ARGSWT=2                                                  INITL  318
                      GO TO 100                                                 INITL  319
                C                                                               INITL  320
     75         C     Y DECODE                                                  INITL  321
                C                                                               INITL  322
                  236 CONTINUE                                                  INITL  323
                      ARGSWT=3                                                  INITL  324
                      GO TO 100                                                 INITL  325
     80         C                                                               INITL  326
                C     MIN, INCREMENT SETTINGS                                   INITL  327
                C                                                               INITL  328
                  240 CONTINUE                                                  INITL  329
                      NAMVAL=I+ARGSWT-1                                         INITL  330
     85               IF(ITABLE(NAMVAL) .NE. NOTSET) GO TO 511                  INITL  331
                      ISWTCH=2                                                  INITL  332
                      GO TO 100                                                 INITL  333
                C                                                               INITL  334
                C     SET ARGUMENT VALUE                                        INITL  335
     90         C                                                               INITL  336
                  300 CONTINUE                                                  INITL  337
                C                                                               INITL  338
                C     CHECK FOR MISSING ARGUMENT                                INITL  339
                C                                                               INITL  340
     95               IF(ARG .EQ. 0) GO TO 512                                  INITL  341
                C                                                               INITL  342
                C     ACCEPT                                                    INITL  343
                C                                                               INITL  344
                      ITABLE(NAMVAL)=ARG                                        INITL  345
    100               ISWTCH=1                                                  INITL  346
                      I=NAMVAL                                                  INITL  347
                      IF(I .GT. 2) I=2                                          INITL  348
                      GO TO (310,320),I                                         INITL  349
                C                                                               INITL  350
    105         C     CHECK OLD ELEMENT NAME                                    INITL  351
                C                                                               INITL  352
                  310 CONTINUE                                                  INITL  353
                      INDEXO=ITABLE(1)                                          INITL  354
                      CALL SEARCH(ELETAB,1,INDEXO)                              INITL  355
    110               IF(INDEXO .EQ. 0) GO TO 521                               INITL  356
                      GO TO 100                                                 INITL  357
                C                                                               INITL  358
                C     GET ARGUMENT VALUE                                        INITL  359
                C                                                               INITL  360
```

```
      320 CONTINUE                                                          INITL  361
          DECODE(10,1000,TABLE(NAMVAL)) TABLE(NAMVAL)                       INITL  362
          IF(ACCHK) GO TO 522                                               INITL  363
          GO TO 100                                                         INITL  364
C                                                                           INITL  365
C     CHECK FOR OLD ELEMENT NAME                                            INITL  366
C                                                                           INITL  367
      400 CONTINUE                                                          INITL  368
          IF(INDEXO .EQ. NOTSET) GO TO 513                                  INITL  369
C                                                                           INITL  370
C     STORE NEW VALUES IN AXIS TABLE                                        INITL  371
C     ITABLE     1          2          3          4          5             INITL  372
C     CONTENTS  NAME       XMIN       XINCR      YMIN       YINCR         INITL  373
C                                                                           INITL  374
          DO 410 I=2,5                                                      INITL  375
              IF(ITABLE(I) .EQ. NOTSET) GO TO 410                           INITL  376
              CALL DMGET(WHOL4+I,ELETAB,COLPTR)                             INITL  377
              CALL DMSET(WHOLC+INDEXO,COLPTR,TABLE(I))                      INITL  378
      410 CONTINUE                                                          INITL  379
      420 CONTINUE                                                          INITL  380
          GO TO 10                                                          INITL  381
C                                                                           INITL  382
C     ERROR IN CARD PROCESSING  **  CARD DELETED  **                        INITL  383
C                                                                           INITL  384
      511 CALL ERRORM(21H(16H ARG ALREADY SET),1,ID)                        INITL  385
          CALL SCAN(CHRPOS,INPUT,ARG)                                       INITL  386
          GO TO 100                                                         INITL  387
      512 CALL ERRORM(44H(39H VAL OR DELIM MISSING OR WRONG ARG NAME),1,    INITL  388
     1   ID)                                                               INITL  389
          GO TO 10                                                          INITL  390
      513 CALL ERRORM(24H(19H OLD NAME NOT GIVEN),3,ID)                     INITL  391
          GO TO 10                                                          INITL  392
C                                                                           INITL  393
C     ERROR IN ARGUMENT VALUE  **  CARD DELETED  **                        INITL  394
C                                                                           INITL  395
      521 CALL ERRORM(24H(19H OLD NAME NOT FOUND),1,ID)                     INITL  396
          GO TO 10                                                          INITL  397
      522 CALL ERRORM(21H(16H BAD NUMERIC ARG),2,ID)                        INITL  398
          ITABLE(NAMVAL)=NOTSET                                             INITL  399
          GO TO 100                                                         INITL  400
 1000 FORMAT(E10.0)                                                         INITL  401
          END                                                               INITL  402
```

```
      SUBROUTINE INITEL                                                      INITL  403
      INTEGER COLPTR,ELETAB,FUNCTB                                           INITL  404
      INTEGER CHRPOS,ZTRUNC,ARG                                              INITL  405
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50              INITL  406
      LOGICAL RDCHK                                                          INITL  407
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50     INITL  408
      COMMON /ERROR/ RDCHK                                                   INITL  409
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                              INITL  410
      COMMON /LASTPOS/ CHRPOS                                                INITL  411
      DIMENSION TABLE(3),ITABLE(3),INPUT(80),NAMARG(4)                       INITL  412
      EQUIVALENCE(TABLE,ITABLE)                                              INITL  413
      DATA (NAMARG(I),I=1,4) /1LO,1LN,1LT,1LF/                               INITL  414
      DATA ID,NOTSET /6HINITEL,1H /                                          INITL  415
C                                                                            INITL  416
C     SUBROUTINE INITIALIZES ELEMENTS                                        INITL  417
C     ARGUMENTS ARE OLD=NAME  NEW=NAME  TOLERANCE=NUMBER                     INITL  418
C                   FINISH                                                   INITL  419
C                                                                            INITL  420
C     SET COUNTERS, ARRAYS, AND SWITCHES                                     INITL  421
C                                                                            INITL  422
   10 CONTINUE                                                               INITL  423
      INDEXO=NOTSET                                                          INITL  424
      CHRPOS=1                                                               INITL  425
      ISWTCH=1                                                               INITL  426
      RDCHK=.FALSE.                                                          INITL  427
      DO 20 I=1,3                                                            INITL  428
      ITABLE(I)=NOTSET                                                       INITL  429
   20 CONTINUE                                                               INITL  430
C                                                                            INITL  431
C     READ INITIALIZATION CARD                                               INITL  432
C                                                                            INITL  433
      CALL READCD(INPUT)                                                     INITL  434
C                                                                            INITL  435
C     DECODE ARGUMENT NAME OR VALUE                                          INITL  436
C                                                                            INITL  437
  100 CONTINUE                                                               INITL  438
      CALL SCAN(CHRPOS,INPUT,ARG)                                            INITL  439
      IF(ARG .EQ. 0) GO TO 400                                               INITL  440
      GO TO (200,300),ISWTCH                                                 INITL  441
C                                                                            INITL  442
C     SELECT ARGUMENT                                                        INITL  443
C                                                                            INITL  444
  200 CONTINUE                                                               INITL  445
      ARG=ZTRUNC(ARG,1)                                                      INITL  446
      DO 210 I=1,3                                                           INITL  447
      NAMVAL=I                                                               INITL  448
      IF(ARG .EQ. NAMARG(I)) GO TO 220                                       INITL  449
  210 CONTINUE                                                               INITL  450
      IF(ARG .EQ. NAMARG(4)) RETURN                                          INITL  451
      IF(ARG .EQ. 0) GO TO 400                                               INITL  452
      GO TO 512                                                              INITL  453
C                                                                            INITL  454
C     CHECK FOR PREVIOUS SETTING OF THIS ARGUMENT                            INITL  455
C                                                                            INITL  456
  220 CONTINUE                                                               INITL  457
      IF(ITABLE(NAMVAL) .NE. NOTSET) GO TO 511                               INITL  458
      ISWTCH=2                                                               INITL  459
```

```
                      GO TO 100                                                  INITL  460
                C                                                                INITL  461
60              C     SET ARGUMENT VALUE                                         INITL  462
                C                                                                INITL  463
                  300 CONTINUE                                                   INITL  464
                C                                                                INITL  465
                C     CHECK FOR MISSING ARGUMENT                                 INITL  466
65              C                                                                INITL  467
                      IF(ARG .EQ. 0) GO TO 512                                   INITL  468
                C                                                                INITL  469
                C     ACCEPT                                                     INITL  470
                C                                                                INITL  471
70                    ITABLE(NAMVAL)=ARG                                         INITL  472
                      ISWTCH=1                                                   INITL  473
                      GO TO (310,320,330),NAMVAL                                 INITL  474
                C                                                                INITL  475
                C     PERFORM ERROR CHECKS ON ARGUMENT                           INITL  476
75              C                                                                INITL  477
                C     CHECK OLD ELEMENT NAME                                     INITL  478
                C                                                                INITL  479
                  310 CONTINUE                                                   INITL  480
                      INDEXO=ITABLE(1)                                           INITL  481
80                    CALL SEARCH(ELETAB,1,INDEXO)                               INITL  482
                      IF(INDEXO .EQ. 0) GO TO 521                                INITL  483
                      GO TO 100                                                  INITL  484
                C                                                                INITL  485
                C     CHECK NEW ELEMENT NAME                                     INITL  486
85              C                                                                INITL  487
                  320 CONTINUE                                                   INITL  488
                      INDEXN=ITABLE(2)                                           INITL  489
                      CALL SEARCH(ELETAB,1,INDEXN)                               INITL  490
                      IF(INDEXN .NE. 0) GO TO 522                                INITL  491
90                    GO TO 100                                                  INITL  492
                C                                                                INITL  493
                C     CHECK NEW TOLERANCE                                        INITL  494
                C                                                                INITL  495
                  330 CONTINUE                                                   INITL  496
95                    DECODE(10,1000,TABLE(3))TABLE(3)                           INITL  497
                      IF(ADDMK) GO TO 523                                        INITL  498
                      TABLE(3)=ABS(TABLE(3))                                     INITL  499
                      GO TO 100                                                  INITL  500
                C                                                                INITL  501
100             C     STORE NEW VALUES IN ELETAB                                 INITL  502
                C            ITABLE     1          2          3                 INITL  503
                C           CONTENTS OLD NAME  NEW NAME  TOLERANCE               INITL  504
                C     CHECK FOR PRESENCE OF OLD                                  INITL  505
                C                                                                INITL  506
105               400 CONTINUE                                                   INITL  507
                      IF(INDEXO .EQ. NOTSET) GO TO 524                           INITL  508
                C                                                                INITL  509
                C     SET NEW TOLERANCE                                          INITL  510
                C                                                                INITL  511
110                   IF(ITABLE(3) .EQ. NOTSET) GO TO 410                        INITL  512
                      CALL DMGET(WHOLE,ELETAB,COLPTR)                            INITL  513
                      CALL DMSET(WHOLE+INDEXO,COLPTR,ITABLE(3))                  INITL  514
                C                                                                INITL  515
                C     SET NEW NAME                                               INITL  516
```

```
      C                                                                 INITL  517
        410 CONTINUE                                                    INITL  518
            IF(ITABLE(2) .EQ. NOTSET) GO TO 10                          INITL  519
            CALL DMGET(WHOL3,ELETAB,COLPTR)                             INITL  520
            CALL DMSET(WHOL0+INDEX0,COLPTR,ITABLE(2))                   INITL  521
      C                                                                 INITL  522
      C     RE SORT ELETAB                                              INITL  523
      C                                                                 INITL  524
            CALL SORTAB(ELETAB,1)                                       INITL  525
            GO TO 10                                                    INITL  526
      C                                                                 INITL  527
      C     ERROR IN CARD PROCESSING  **  CARD DELETED  **              INITL  528
      C                                                                 INITL  529
        511 CALL ERRORM(21H(16H ARG ALREADY SET),1,ID)                  INITL  530
            CALL SCAN(CHRPOS,INPUT,ARG)                                 INITL  531
            GO TO 100                                                   INITL  532
        512 CALL ERRORM(44H(39H VAL OR DELIM MISSING OR WRONG ARG NAME),1,INITL  533
           1  ID)                                                       INITL  534
            GO TO 10                                                    INITL  535
      C                                                                 INITL  536
      C     ERROR IN ARGUMENT VALUE  **  CARD DELETED  **               INITL  537
      C                                                                 INITL  538
        521 CALL ERRORM(24H(19H OLD NAME NOT FOUND),1,ID)               INITL  539
            GO TO 10                                                    INITL  540
        522 CALL ERRORM(21H(16H NEW NAME EXISTS),1,ID)                  INITL  541
            GO TO 10                                                    INITL  542
        523 CALL ERRORM(21H(16H BAD NUMERIC ARG),2,ID)                  INITL  543
            ITABLE(3)=NOTSET                                            INITL  544
            GO TO 100                                                   INITL  545
        524 CALL ERRORM(22H(17H OLD NAME MISSING),3,ID)                 INITL  546
            GO TO 10                                                    INITL  547
       1000 FORMAT(E10.0)                                               INITL  548
            END                                                         INITL  549
```

```
      SUBROUTINE LISTAX                                                 INITL   550
      INTEGER ELETAB,FUNCTB,AXSPTR(4)                                   INITL   551
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50         INITL   552
      COMMON /COMFCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 INITL   553
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                         INITL   554
      DIMENSION AXSCAL(4)                                               INITL   555
C                                                                       INITL   556
C     SUBROUTINE LISTS AXES                                             INITL   557
C                                                                       INITL   558
C     OUTPUT FIRST HEADER                                               INITL   559
C                                                                       INITL   560
      WRITE(6,1001)                                                     INITL   561
C                                                                       INITL   562
C     OUTPUT SECOND HEADER                                              INITL   563
C                                                                       INITL   564
      WRITE(6,1002)                                                     INITL   565
C                                                                       INITL   566
C     GENERATE TABLE OF ELEMENT NAMES AND AXES SCALE SETTINGS           INITL   567
C                                                                       INITL   568
      CALL DMGET(WHOL1,ELETAB,NOOFEL)                                   INITL   569
      DO 10 I=1,4                                                       INITL   570
         CALL DMGET(WHOL5+I,ELETAB,AXSPTR(I))                           INITL   571
   10    CONTINUE                                                       INITL   572
      CALL DMGET(WHOL3,ELETAB,NAMPTR)                                   INITL   573
      DO 50 I=1,NOOFEL                                                  INITL   574
      CALL DMGET(WHOL0+I,NAMPTR,NAME)                                   INITL   575
      DO 30 J=1,4                                                       INITL   576
         CALL DMGET(WHOL0+I,AXSPTR(J),AXSCAL(J))                        INITL   577
   30    CONTINUE                                                       INITL   578
      XMAX=AXSCAL(1)+XSCAL*AXSCAL(2)                                    INITL   579
      YMAX=AXSCAL(3)+YSCAL*AXSCAL(4)                                    INITL   580
      WRITE(6,1005) NAME,AXSCAL(1),AXSCAL(2),XMAX,AXSCAL(3),AXSCAL(4),  INITL   581
     *  YMAX                                                            INITL   582
   50 CONTINUE                                                          INITL   583
      RETURN                                                            INITL   584
 1001 FORMAT(//,15X,11HAXES TABLES,//)                                  INITL   585
 1002 FORMAT(5X,4HNAME,10X,7HMINIMUM,6X,9HINCREMENT,5X,7HMAXIMUM)       INITL   586
 1005 FORMAT(2X,A10,2X,1HX,3X,G11.5,2X,G11.5,3X,G11.5,/,                 INITL   587
     1  14X,1HY,3X,G11.5,2X,G11.5,3X,G11.5)                            INITL   588
      END                                                               INITL   589
```

```
      SUBROUTINE LISTEL                                                 INITL     590
      INTEGER COLPTR(3),ELEM(3)                                         INITL     591
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50         INITL     592
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 INITL     593
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                         INITL     594
      DATA NS,FUNCTN,REGULR,SURELE /6HNOTSET,3HFUN,3HREG,3HSUR/         INITL     595
C                                                                       INITL     596
C     SUBROUTINE LISTS ELEMENT TABLES                                   INITL     597
C                                                                       INITL     598
C     OUTPUT FIRST HEADER                                               INITL     599
C                                                                       INITL     600
      WRITE(6,1005)                                                     INITL     601
C                                                                       INITL     602
C     OUTPUT SECOND HEADER                                              INITL     603
C                                                                       INITL     604
      WRITE(6,1006)                                                     INITL     605
C                                                                       INITL     606
C     OUTPUT ELEMENT LIST                                               INITL     607
C                                                                       INITL     608
      CALL DMGET(WHOL1,ELETAB,NOOFEL)                                   INITL     609
      DO 10 I=1,3                                                       INITL     610
         CALL DMGET(WHOL2+I,ELETAB,COLPTR(I))                           INITL     611
   10    CONTINUE                                                       INITL     612
      DO 50 I=1,NOOFEL                                                  INITL     613
      DO 20 J=1,3                                                       INITL     614
         CALL DMGET(WHOL0+I,COLPTR(J),ELEM(J))                          INITL     615
   20    CONTINUE                                                       INITL     616
      IF(ELEM(2) .LT. 500) GO TO 35                                     INITL     617
      IF(ELEM(2) .LT. 600) GO TO 40                                     INITL     618
      OUT=SURELS                                                        INITL     619
      GO TO 45                                                          INITL     620
   35 OUT=REGULR                                                        INITL     621
      GO TO 45                                                          INITL     622
   40 OUT=FUNCTN                                                        INITL     623
   45 CONTINUE                                                          INITL     624
      WRITE(6,1007) ELEM(1),ELEM(3),OUT                                 INITL     625
   50 CONTINUE                                                          INITL     626
      RETURN                                                            INITL     627
 1000 FORMAT(G10.2)                                                     INITL     628
 1005 FORMAT(//,10X,14HELEMENT TABLES,//)                               INITL     629
 1006 FORMAT(5X,4HNAME,7X,9HTOLERANCE,6X,4HTYPE,//)                     INITL     630
 1007 FORMAT(2X,A10,3X,G11.5,5X,A3)                                     INITL     631
      END                                                               INITL     632
```

```
      PROGRAM PS1MAIN                                                    INITL  633
      INTEGER CHRPOS,ZTRUNC,ARG                                          INITL  634
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50          INITL  635
      LOGICAL POINTS,FRSTPS,ROCHK,LISTSF                                 INITL  636
      DIMENSION INPUT(80)                                                INITL  637
      DIMENSION NAMAG1(4),NAMAG2(4),NAMAG3(3),NAMAG4(2),ITABLE(3)        INITL  638
      EQUIVALENCE(RLARG,ARG)                                             INITL  639
      COMMON/COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50  INITL  640
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                         INITL  641
      COMMON /LASTPOS/ CHRPOS                                            INITL  642
      COMMON /PLTSZ/ VFACTR                                              INITL  643
      COMMON /ERROR/ ROCHK                                               INITL  644
      COMMON /INITSW/ LISTSF,POINTS,NOOFTP                               INITL  645
      DATA FRSTPS,NOTSET /.TRUE.,1H /                                    INITL  646
      DATA (NAMAG1(I),I=1,4) /1LI,1LF,1LP,1LL/                           INITL  647
      DATA (NAMAG2(I),I=1,4) /1LT,1LE,1LA,1LR/                           INITL  648
      DATA (NAMAG3(I),I=1,3) /1LE,1LS,1LA/                               INITL  649
      DATA (NAMAG4(I),I=1,2) /1LL,1LN/                                   INITL  650
      DATA ID /7HPS1MAIN/                                                INITL  651
C                                                                        INITL  652
C     PROGRAM DRIVES INITIALIZATION ROUTINES                             INITL  653
C     IO UNITS AND ELEMENTS MAY BE DEFAULTED                             INITL  654
C                                                                        INITL  655
C     ARGUMENTS ARE                                                      INITL  656
C     LIST  (ELEMENTS,SURFACES,AXES)                                     INITL  657
C     INITIALIZE  (ELEMENTS,TAPES,AXES,READ-IN)                          INITL  658
C     POINTS (LIST,NOLIST)                                               INITL  659
C     FINISH INITIALIZATION                                              INITL  660
C                                                                        INITL  661
C                                                                        INITL  662
C     FIRST PASS INITIALIZATION                                          INITL  663
C                                                                        INITL  664
      WRITE(6,1000)                                                      INITL  665
      IF(FRSTPS) GO TO 10                                                INITL  666
      GO TO 30                                                           INITL  667
   10 CONTINUE                                                           INITL  668
C                                                                        INITL  669
C        INITIALIZE ELEMENT AND FUNCTION TABLES                          INITL  670
      CALL DEFELE                                                        INITL  671
      CALL DEFFUN                                                        INITL  672
      FRSTPS=.FALSE.                                                     INITL  673
C                                                                        INITL  674
C     INITIALIZE SWITCHES AND ARRAYS                                     INITL  675
C                                                                        INITL  676
   30 CONTINUE                                                           INITL  677
      CHRPOS=1                                                           INITL  678
      ROCHK=.FALSE.                                                      INITL  679
C                                                                        INITL  680
C     READ CARD                                                          INITL  681
C                                                                        INITL  682
   40 CONTINUE                                                           INITL  683
      CALL READCD(INPUT)                                                 INITL  684
C                                                                        INITL  685
C     GET ARGUMENT 1                                                     INITL  686
C                                                                        INITL  687
  100 CONTINUE                                                           INITL  688
      CALL SCAN(CHRPOS,INPUT,ARG)                                        INITL  689
```

```
      ARG=ZTRUNC(ARG,1)                                                    INITL  690
C                                                                          INITL  691
C     DETERMINE ARG1 (LIST,INITIALIZATION,FINISH,POINTS)                   INITL  692
C                                                                          INITL  693
  200 CONTINUE                                                             INITL  694
      DO 210 I=1,4                                                         INITL  695
      IF(ARG .EQ. NAMAG1(I)) GO TO (300,320,330,340),I                     INITL  696
  210 CONTINUE                                                             INITL  697
      GO TO 511                                                            INITL  698
C                                                                          INITL  699
C     ARG1=INITIALIZE, DETERMINE ARG2 (ELEMENTS,TAPES,AXES,READ-IN)        INITL  700
C                                                                          INITL  701
  300 CONTINUE                                                             INITL  702
      CALL SCAN(CHRPOS,INPUT,ARG)                                          INITL  703
      ARG=ZTRUNC(ARG,1)                                                    INITL  704
      DO 310 I=1,4                                                         INITL  705
      IF (ARG .EQ. NAMAG2(I)) GO TO (312,314,316,318),I                    INITL  706
  310 CONTINUE                                                             INITL  707
      GO TO 512                                                            INITL  708
  312 CALL SCAN(CHRPOS,INPUT,ARG)                                          INITL  709
      IF(ARG .EQ. 0) GO TO 518                                             INITL  710
      DECODE(10,1001,RLARG) RLARG                                          INITL  711
      IF (.NOT.RDCHK) GO TO 313                                            INITL  712
      RDCHK = .FALSE.                                                      INITL  713
      GO TO 519                                                            INITL  714
  313 CONTINUE                                                             INITL  715
      NOOFTP=ABS(RLARG)                                                    INITL  716
      IF(NOOFTP .GT. 6) NOOFTP=6                                           INITL  717
      GO TO 30                                                             INITL  718
  314 CALL ELMENT(INPUT)                                                   INITL  719
      GO TO 30                                                             INITL  720
  316 CALL AXES(INPUT)                                                     INITL  721
      GO TO 30                                                             INITL  722
  318 CALL FOSET(INPUT)                                                    INITL  723
      GO TO 30                                                             INITL  724
C                                                                          INITL  725
C        ARG1 WAS FINISH OR FACTOR                                         INITL  726
C        LET ARG2 DETERMINT VERB          INIT=FIN, NUM=FACTOR             INITL  727
C                                                                          INITL  728
  320 CONTINUE                                                             INITL  729
      CALL SCAN(CHRPOS,INPUT,ARG)                                          INITL  730
      IF (ZTRUNC(ARG,1) .EQ. NAMAG1(1)) GO TO 600                          INITL  731
C                                                                          INITL  732
C        DETERMINE IF NUMERIC                                              INITL  733
      DECODE(10,1001,ARG) RLARG                                            INITL  734
      IF (.NOT.RDCHK) GO TO 325                                            INITL  735
      RDCHK = .FALSE.                                                      INITL  736
      GO TO 514                                                            INITL  737
  325 VFACTR = RLARG                                                       INITL  738
      GO TO 30                                                             INITL  739
C                                                                          INITL  740
C     ARG 1 WAS POINTS  DETERMINE ARG 2  (LIST,NOLIST)                     INITL  741
C                                                                          INITL  742
  330 CONTINUE                                                             INITL  743
      CALL SCAN(CHRPOS,INPUT,ARG)                                          INITL  744
      ARG=ZTRUNC(ARG,1)                                                    INITL  745
      DO 335 I=1,2                                                         INITL  746
```

```
      IF(ARG .EQ. NAMAG4(I)) GO TO (337,338),I                          INITL  747
  335 CONTINUE                                                          INITL  748
      GO TO 517                                                         INITL  749
  337 POINTS=.TRUE.                                                     INITL  750
      GO TO 30                                                          INITL  751
  338 POINTS=.FALSE.                                                    INITL  752
      GO TO 30                                                          INITL  753
C                                                                       INITL  754
C     ARG 1 WAS LIST,  DETERMINE ARG 2  (AXES,ELEMENTS,SURFACES)        INITL  755
C                                                                       INITL  756
  340 CONTINUE                                                          INITL  757
      DO 342 I=1,3                                                      INITL  758
      ITABLE(I)=NOTSET                                                  INITL  759
  342 CONTINUE                                                          INITL  760
  343 CONTINUE                                                          INITL  761
      CALL SCAN(CHRPOS,INPUT,ARG)                                       INITL  762
      ARG=ZRUNC(ARG,1)                                                  INITL  763
      IF(ARG .EQ. 0) GO TO 348                                          INITL  764
      DO 344 I=1,3                                                      INITL  765
      NAMVAL=I                                                          INITL  766
      IF(ARG .EQ. NAMAG3(I)) GO TO 346                                  INITL  767
  344 CONTINUE                                                          INITL  768
      GO TO 516                                                         INITL  769
C                                                                       INITL  770
C     ACCEPT ARGUMENT                                                   INITL  771
C                                                                       INITL  772
  346 CONTINUE                                                          INITL  773
      ITABLE(NAMVAL)=1                                                  INITL  774
      GO TO 343                                                         INITL  775
C                                                                       INITL  776
C     EVALUATE LISTS                                                    INITL  777
C                                                                       INITL  778
  348 CONTINUE                                                          INITL  779
      IF(ITABLE(1) .EQ. 1) CALL LISTEL                                  INITL  780
      IF(ITABLE(2) .EQ. 1) LISTSF=.TRUE.                                INITL  781
      IF(ITABLE(3) .EQ. 1) CALL LISTAX                                  INITL  782
      GO TO 30                                                          INITL  783
C                                                                       INITL  784
C     ERROR IN PROCESSING CARD  **  CARD DELETED  **                    INITL  785
C                                                                       INITL  786
  511 CALL ERRORM(34H(29H ARG1=INIT,LIST,FINISH,POINTS),1,ID)           INITL  787
      GO TO 30                                                          INITL  788
  512 CALL ERRORM(25H(20H ARG2=TAPE,ELEM,AXES),1,ID)                    INITL  789
      GO TO 30                                                          INITL  790
  514 CALL ERRORM(35H(30H ARG2=INIT,TERMINATION ASSUMED),2,ID)          INITL  791
      GO TO 600                                                         INITL  792
  516 CALL ERRORM(30H(25H ARG2,3,4 =ELEM,SURF,AXES),1,ID)               INITL  793
      GO TO 30                                                          INITL  794
  517 CALL ERRORM(22H(17H ARG2=LIST,NOLIST),1,ID)                       INITL  795
      GO TO 30                                                          INITL  796
  518 CALL ERRORM(26H(21H BAD NUMERIC ARGUMENT),1,ID)                   INITL  797
      GO TO 30                                                          INITL  798
  600 CONTINUE                                                          INITL  799
 1000 FORMAT(/////,20X,20HINITIALIZATION PHASE,///)                     INITL  800
 1001 FORMAT(E10.0)                                                     INITL  801
      END                                                               INITL  802
```

```
      SUBROUTINE RDSET(INPUT)                                          INITL  803
C                                                                      INITL  804
C        THIS ROUTINE IS USED TO INITIALIZE THE NUMBER OF VARIABLES TO INITL  805
C        READ IN AND ALSO THE FORMAT UNDER WHICH TO READ THEM.         INITL  806
C                                                                      INITL  807
      COMMON/LASTPOS/ LPS                                              INITL  808
      COMMON/INPTRC/ V(100),NOOFWD,NOOFRC,RDFMT(24)                    INITL  809
      DIMENSION IARG(2),INPUT(1)                                       INITL  810
      COMMON /ERROR/ CHECK                                             INITL  811
      LOGICAL CHECK                                                    INITL  812
      INTEGER ZTRUNC                                                   INITL  813
      DATA (IARG(I),I=1,2) /1LV,1LF/                                   INITL  814
      DATA ID/8HRDSET/                                                 INITL  815
      VAL1 = NOOFWD                                                    INITL  816
C                                                                      INITL  817
C        SCAN FOR VNUM OR FCARD INDICATOR                              INITL  818
    5 CALL SCAN(LPS,INPUT,IWORD)                                       INITL  819
      IF (IWORD .EQ. 0) GO TO 45                                       INITL  820
      IWORC = ZTRUNC(IWORD,1)                                          INITL  821
      DO 10 I=1,2                                                      INITL  822
      IF (IWORD .EQ. IARG(I)) GO TO (30,40),I                          INITL  823
   10 CONTINUE                                                         INITL  824
      GO TO 9991                                                       INITL  825
C                                                                      INITL  826
C        INITIALIZE NUMBER OF VARIABLES                                INITL  827
   30 CALL SCAN(LPS,INPUT,IWORD)                                       INITL  828
      IF (IWORD .EQ. 0) GO TO 45                                       INITL  829
C                                                                      INITL  830
C        DECODE NUMERIC                                                INITL  831
      DECODE(10,35,IWORD) VAL1                                         INITL  832
   35 FORMAT(E10.0)                                                    INITL  833
      IF (.NOT.CHECK) GO TO 5                                          INITL  834
   37 CHECK =.FALSE.                                                   INITL  835
      GO TO 9991                                                       INITL  836
C                                                                      INITL  837
C        INITIALIZE NUMBER OF FORMAT CARDS                             INITL  838
   40 CALL SCAN(LPS,INPUT,IWORD)                                       INITL  839
      IF (IWORD .EQ. 0) GO TO 45                                       INITL  840
C                                                                      INITL  841
C        DECODE NUMERIC                                                INITL  842
      DECODE(10,35,IWORD) VAL2                                         INITL  843
      IF (CHECK) GO TO 37                                              INITL  844
      GO TO 5                                                          INITL  845
C                                                                      INITL  846
C        DECIDE WHETHER TO INITL READ FMT                              INITL  847
   45 IF (VAL2 .LE. 0. .OR. VAL2 .GT. 3.) GO TO 9992                   INITL  848
      IF (VAL1 .LE. 1.) GO TO 9993                                     INITL  849
      NWRD = VAL2*8.                                                   INITL  850
      NOOFWD = VAL1                                                    INITL  851
      READ(5,50) (RDFMT(I),I=1,NWRD)                                   INITL  852
   50 FORMAT(8A10)                                                     INITL  853
      WRITE(6,60) (RDFMT(I),I=1,NWRD)                                  INITL  854
   60 FORMAT(* *,8A10)                                                 INITL  855
      RETURN                                                           INITL  856
C        ERROR CALLS                                                   INITL  857
 9991 CALL ERROFM(22H(17H ILLEGAL ARGUMENT),2,ID)                      INITL  858
      GO TO 5                                                          INITL  859
```

```
      9992 CALL ERROR4(38H(33H INCORRECT NUMBER OF FORMAT CARDS),-1,ID)      INITL    860
      9993 CALL ERROR4(33H(28H ILLEGAL NUMBER OF VARIABLES),-1,ID)           INITL    861
60          END                                                             INITL    862
```

```
1                SUBROUTINE READCO(INPUT)                                          INITL    863
                 DIMENSION INPUT(80)                                               INITL    864
                 READ(5,100) (INPUT(I),I=1,80)                                     INITL    865
                 IF(EOF(5) .NE. 0.) GO TO 600                                      INITL    866
5                WRITE(6,101) (INPUT(I),I=1,80 )                                   INITL    867
                 RETURN                                                            INITL    868
           600   CALL ERRORM(29H(24H END OF FILE ENCOUNTERED),-3,6HREADRC)          INITL    869
           100   FORMAT(80A1)                                                      INITL    870
           101   FORMAT(10X,80A1)                                                  INITL    871
10               END                                                               INITL    872
```

```
      SUBROUTINE SCAN(IFROM,ARRAY,IWORD)                                INITL  873
C-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*- INITL  874
C                                                                       INITL  875
C        THIS SUBROUTINE SCANS AN INPUT ARRAY FOR A FIELD OF CONTINUOUS  INITL  876
C        CHARACTERS AND RETURNS THE LEFTMOST TEN LEFT JUSTIFIED AND BLAN INITL  877
C        IFROM  -  THE POINT WHERE SCANNING BEGINS, PASSED IN           INITL  878
C               -  ONE PAST THE LAST POSITION SCANNED, PASSED OUT       INITL  879
C        ARRAY  -  ARRAY INTO WHICH THE INPUT RECORD HAS BEEN READ UNDE INITL  880
C        IWORD  -  WORD CONTAINING SCANNED CHARACTERS                   INITL  881
C-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*- INITL  882
      INTEGER ARRAY(1)                                                  INITL  883
      DATA IBLK/10H          /                                          INITL  884
      DATA ICOMMA/10H,         /                                        INITL  885
      DATA IEQ/10H=          /                                          INITL  886
      IF (IFROM .GT. 80) GO TO 15                                       INITL  887
C                                                                       INITL  888
C        FIND FIRST NON-BLANK,NON-COMMA CHARACTER                       INITL  889
      DO 10 J=IFROM,80                                                  INITL  890
      IF (ARRAY(J) .NE. IBLK .AND. ARRAY(J) .NE. ICOMMA                 INITL  891
     X .AND. ARRAY(J) .NE. IEQ) GO TO 20                                INITL  892
   10 CONTINUE                                                          INITL  893
C                                                                       INITL  894
C        IF IWORD EQUALS ZERO THEN NO CHARACTERS REMAIN                 INITL  895
   15 IWORD = 0                                                         INITL  896
      RETURN                                                            INITL  897
C                                                                       INITL  898
C        FIND NEXT BLANK OR COMMA DELIMITER                             INITL  899
   20 DO 30 K=J,80                                                      INITL  900
      IF (ARRAY(K) .EQ. IBLK .OR. ARRAY(K) .EQ. ICOMMA                  INITL  901
     X .OR. ARRAY(K) .EQ. IEQ) GO TO 40                                 INITL  902
   30 CONTINUE                                                          INITL  903
C                                                                       INITL  904
C        BLANK DELIMITED CHARACTERS OCCUPY POSITIONS J TO K-1           INITL  905
   40 IFROM = K+1                                                       INITL  906
      L = K-1                                                           INITL  907
      K = K-J                                                           INITL  908
      IWORD = IBLK                                                      INITL  909
C                                                                       INITL  910
C        USE ONLY THE TEN LEFTMOST CHARACTERS WHEN BUILDING WORD        INITL  911
      IF (K .GT. 10) L = J+9                                            INITL  912
      ENCODE (10,55,IWORD)(ARRAY(K),K=J,L)                              INITL  913
   55 FORMAT(10A1)                                                      INITL  914
      RETURN                                                            INITL  915
      END                                                               INITL  916
```

```
      SUBROUTINE STRTAB(TABPOS,STRTWD,NOFWRD,NAMPTR,FORMT,DIGPTR,CODE)
C
C     ROUTINE STORE DEFAULT NAME AND CODE DIGIT IN TABLE BEADS
C
      INTEGER TABPOS,FORMT,DIGPTR,CODE,CODIG,STRTWD
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50
C
      CODIG=CODE
      I=STRTWD-1
      LSTWRD=I+NOFWRD
    5 CONTINUE
      I=I+1
      IF(I .GT. LSTWRD) RETURN
      ICOMP=WHOL0+TABPOS
           IF(I .GT. 9) GO TO 10
           ENCODE(10,1000,NO) I
           GO TO 30
   10      IF(I .GT. 99) GO TO 20
           ENCODE(10,1001,NO) I
           GO TO 30
   20      ENCODE(10,1002,NO) I
   30      ENCODE(10,FORMT,NAME) NO
           CALL DMSET(ICOMP,NAMPTR,NAME)
           CODIG=CODIG+1
           CALL DMSET(ICOMP,DIGPTR,CODIG)
           TABPOS=TABPOS+1
           GO TO 5
C
C
 1000 FORMAT(I1,9X)
 1001 FORMAT(I2,8X)
 1002 FORMAT(I3,7X)
      END
```

```
      OVERLAY(2,0)                                                      DEF        2
      SUBROUTINE LISTSF                                                 DEF        3
      INTEGER OFF,OS,UNU,ELETAB,FUNCTB,TOLPTR,SURFTB                    DEF        4
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                         DEF        5
      COMMON /SFTAB/ SURFTB(10,11)                                      DEF        6
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5                       DEF        7
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5              DEF        8
      COMMON /CONSTN/ BOVERC,NOOFDG                                     DEF        9
      DIMENSION LOCTAB(10)                                              DEF       10
      DATA NOTSET,NS /1H ,6HNOTSET/                                     DEF       11
      DATA OFF,OS,UNU/999.,5H OFF ,5H UNU /                             DEF       12
C                                                                       DEF       13
C     SUBROUTINE LISTS SURFACE TABLES                                   DEF       14
C                                                                       DEF       15
C     OUTPUT FIRST HEADER                                               DEF       16
C                                                                       DEF       17
      WRITE(6,1005)                                                     DEF       18
C                                                                       DEF       19
C     LOCATE SURFACES IN ELEMENT TABLE                                  DEF       20
C                                                                       DEF       21
      CALL SURFSH(ELETAB,LOCTAB)                                        DEF       22
C                                                                       DEF       23
C     OUTPUT SURFACE ELEMENTS                                           DEF       24
C                                                                       DEF       25
      CALL DMGET(WHOL3,ELETAB,NAMPTR)                                   DEF       26
      CALL DMGET(WHOL5,ELETAB,TOLPTR)                                   DEF       27
      DO 10 I=1,10                                                      DEF       28
      J=LOCTAB(I)                                                       DEF       29
      ICOMP=WHOL0+J                                                     DEF       30
      CALL DMGET(ICOMP,NAMPTR,NAME)                                     DEF       31
      CALL DMGET(ICOMP,TOLPTR,TOL)                                      DEF       32
      WRITE(6,1006) NAME,TOL,I                                          DEF       33
   10 CONTINUE                                                          DEF       34
C                                                                       DEF       35
C     OUTPUT NO OF DIGITS HEADER                                        DEF       36
C                                                                       DEF       37
      WRITE(6,1007) NOOFDG                                              DEF       38
C                                                                       DEF       39
C     OUTPUT SURFACE TABLE HEADER                                       DEF       40
C                                                                       DEF       41
      WRITE(6,1008)                                                     DEF       42
C                                                                       DEF       43
C     OUTPUT SURFACE LIST                                               DEF       44
C                                                                       DEF       45
      DO 80 I=1,10                                                      DEF       46
C                                                                       DEF       47
C     CHECK FOR SURFACE UNUSED                                          DEF       48
C                                                                       DEF       49
      IF(SURFTB(I,11) .EQ. NOTSET) GO TO 70                             DEF       50
C                                                                       DEF       51
C     SEARCH SURFACE ENTRIES FOR OFF,NOTSET                             DEF       52
C                                                                       DEF       53
      DO 60 J=1,10                                                      DEF       54
      IF(SURFTB(I,J) .EQ. OFF) GO TO 45                                 DEF       55
      IF(SURFTB(I,J) .EQ. NOTSET) GO TO 50                              DEF       56
      ENCODE(10,1001,LOCTAB(J))SURFTB(I,J)                              DEF       57
      GO TO 60                                                          DEF       58
```

```
      C                                                                 DEF        59
      C     CEF CASE                                                    DEF        60
60    C                                                                 DEF        61
        45  CONTINUE                                                    DEF        62
            LOCTAB(J)=OS                                              DEF        63
            GO TO 60                                                  DEF        64
      C                                                                 DEF        65
65    C     NOTSET CASE                                                 DEF        66
      C                                                                 DEF        67
        50  CONTINUE.                                                   DEF        68
            LOCTAB(J)=UNU                                             DEF        69
        60  CONTINUE                                                    DEF        70
70    C                                                                 DEF        71
      C     OUTPUT LINE                                                 DEF        72
      C                                                                 DEF        73
            WRITE(6,1009) I,(LOCTAB(J),J=1,10),SURFTB(I,11)           DEF        74
            GO TO 80                                                  DEF        75
75    C                                                                 DEF        76
      C     SURFACE NOT USED CASE                                       DEF        77
      C                                                                 DEF        78
        70  CONTINUE                                                    DEF        79
            LOCTAB(1)=NOTSET                                          DEF        80
80          LOCTAB(2)=UNU                                             DEF        81
            WRITE(6,1009) I,(LOCTAB(J),J=1,2)                         DEF        82
        80  CONTINUE                                                    DEF        83
            RETURN                                                    DEF        84
      1000  FORMAT(I2,9X)                                             DEF        85
85    1001  FORMAT(F5.1,5X)                                           DEF        86
      1005  FORMAT(//,10X,14HSURFACE TABLES,//,5X,4HNAME,7X,9HTOLERANCE,6X,  DEF  87
     1  7HSURFACE,//)                                                 DEF        88
      1006  FORMAT(2X,A10,3X,G11.5,8X,I2)                             DEF        89
      1007  FORMAT(//,5X,21HNO OF DIGITS IN V4 = ,I3,//)              DEF        90
90    1008  FORMAT(8H SURFACE,3X,1H1,4X,1H2,4X,1H3,4X,1H4,4X,1H5,4X,1H6,  DEF  91
     1  4X,1H7,4X,1H8,4X,1H9,4X,1H0,3X,5HDIGIT,//)                    DEF        92
      1009  FORMAT(3X,I2,4X,10A5,2X,I2)                               DEF        93
            END                                                       DEF        94
```

```
      PROGRAM PSSMAIN                                                   DEF        95
      LOGICAL LSTSUR,POINTS                                             DEF        96
      INTEGER ELETAB,FUNCTB,SURFTB,TAPENO,UNITNO                        DEF        97
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5                       DEF        98
C                                                                       DEF        99
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5              DEF       100
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                         DEF       101
      COMMON /SFTAB/ SURFTB(10,11)                                      DEF       102
      COMMON /TAPE/ TAPENO,UNITNO                                       DEF       103
      COMMON /CONSTN/ BOVERC,NOOFBG                                     DEF       104
      COMMON /ERROR/ BDCHK                                              DEF       105
      COMMON /INITSW/ LSTSUR,POINTS,NOOFTP                              DEF       106
C                                                                       DEF       107
C                                                                       DEF       108
C                                                                       DEF       109
C     PROGRAM INITIALIZES TABLES ASSOCIATED WITH DATA TAPES             DEF       110
C     SURFACES AND B/C                                                  DEF       111
C                                                                       DEF       112
C                                                                       DEF       113
C     INITIALIZE SURFACE TABLE TO 'NOT USED SETTING                     DEF       114
C                                                                       DEF       115
      CALL SURFST(0)                                                    DEF       116
C                                                                       DEF       117
C     INITIALIZE DATA TAPE                                              DEF       118
C                                                                       DEF       119
      CALL TAPE                                                         DEF       120
      IF(LSTSUR) CALL LISTSF                                            DEF       121
      END                                                               DEF       122
```

```
      SUBROUTINE SURFSH(TABPTR,LOCTAB)                                  DEF     123
C                                                                       DEF     124
      INTEGER TABPTR,DIGPTR                                             DEF     125
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5                       DEF     126
C                                                                       DEF     127
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5              DEF     128
      DIMENSION LOCTAB(1)                                               DEF     129
C                                                                       DEF     130
C     SUBROUTINE LOCATES ALL INSTANCES OF IFOR IN ITABLE AND STORES     DEF     131
C     THESE SUBSCRIPTS IN LOCTAB                                        DEF     132
C                                                                       DEF     133
C     LOOK IS THE SEARCH COLUMN IN ITABLE                               DEF     134
C     ITABSZ IS THE SIZE OF ITABLE                                      DEF     135
C                                                                       DEF     136
      CALL DMGET(WHOL1,TABPTR,NOFEL)                                    DEF     137
      CALL DMGET(WHOL4,TABPTR,DIGPTR)                                   DEF     138
      DO 20 I=1,NOFEL                                                   DEF     139
         CALL DMGET(WHOL0+I,DIGPTR,ITEM)                                DEF     140
         IF(ITEM .GT. 600 .AND. ITEM .LE. 699) GO TO 10                 DEF     141
         GO TO 20                                                       DEF     142
   10    CONTINUE                                                       DEF     143
         J=ITEM-600                                                     DEF     144
         LOCTAB(J)=I                                                    DEF     145
      LOCTAB(J)=I                                                       DEF     146
   20 CONTINUE                                                          DEF     147
      RETURN                                                            DEF     148
      END                                                               DEF     149
```

```
      SUBROUTINE SURFST(SURFNO)                                          DEF     150
      INTEGER SURFTB,SURFNO                                              DEF     151
C                                                                        DEF     152
      COMMON /SFTAB/ SURFTB(10,11)                                       DEF     153
      DATA NOTSET /1H /                                                  DEF     154
C                                                                        DEF     155
C     SUBROUTINE SETS SURFACES TO NOT USED SETTING                       DEF     156
C                                                                        DEF     157
      IF(SURFNO .EQ. 0) GO TO 5                                          DEF     158
      K=L=SURFNO                                                         DEF     159
      GO TO 6                                                            DEF     160
    5 K=1                                                                DEF     161
      L=10                                                               DEF     162
    6 DO 10 J=K,L                                                        DEF     163
      DO 20 I=1,11                                                       DEF     164
      SURFTB(J,I)=NOTSET                                                 DEF     165
   20 CONTINUE                                                           DEF     166
   10 CONTINUE                                                           DEF     167
      RETURN                                                             DEF     168
      END                                                                DEF     169
```

```
      SUBROUTINE TAPE                                                   DEF     170
      COMMON /TAPE/ TAPENO,UNITNO                                       DEF     171
      COMMON /CONSTN/ BOVERC,NOOFDG                                     DEF     172
      COMMON /ERROR/ ROCHK                                              DEF     173
      COMMON /SETAB/ SURFTB(10,11)                                      DEF     174
      LOGICAL ROCHK                                                     DEF     175
      INTEGER CODE,DIGIT,VALUE                                          DEF     176
      INTEGER SURFTB,TAPENO,UNITNO                                      DEF     177
      DIMENSION CODE(10),VALUE(10)                                      DEF     178
      DATA ID /4HTAPE/                                                  DEF     179
C                                                                       DEF     180
C     SUBROUTINE INITIALIZES CONSTANTS FROM TAPE                        DEF     181
C                                                                       DEF     182
C                                                                       DEF     183
C     GET B/C AND NUMBER OF DIGITS IN V4                                DEF     184
C                                                                       DEF     185
      ROCHK=.FALSE.                                                     DEF     186
      READ(UNITNO,1000) BOVERC,NOOFDG                                   DEF     187
      IF(EOF(UNITNO) .NE. 0.) GO TO 502                                 DEF     188
      IF(ROCHK) GO TO 501                                               DEF     189
C                                                                       DEF     190
C     VERIFY B/C AND NOOFDG                                             DEF     191
C                                                                       DEF     192
      IF(NOOFDG .LE. 0 .OR. NOOFDG .GT. 10) GO TO 501                   DEF     193
      IF(BOVERC .EQ. 0.) GO TO 501                                      DEF     194
C                                                                       DEF     195
C     LOOP TO GET SURFACE TABLE CONVERSION CODES                        DEF     196
C     FORMAT OF V4 WORD IS                                              DEF     197
C           DIGIT(NOOFDG),...,DIGIT2,DIGIT1.                            DEF     198
C                                                                       DEF     199
C        TABLE POSITION       DIGIT                                     DEF     200
C          SURFTB(1,*)          1                                      DEF     201
C              .                .                                      DEF     202
C              .                .                                      DEF     203
C              .                .                                      DEF     204
C          SURFTB(NOOFDG,*)     NOOFDG                                 DEF     205
C                                                                       DEF     206
      DO 20 J=1,NOOFDG                                                  DEF     207
         READ(UNITNO,1001) DIGIT,NOOFST                                 DEF     208
         IF(EOF(UNITNO) .NE. 0.) GO TO 502                              DEF     209
         IF(ROCHK) GO TO 503                                            DEF     210
         IF(DIGIT .LE. 0 .OR. DIGIT .GT. 10) GO TO 503                  DEF     211
         IF(NOOFST .LE. 0 .OR. NOOFST .GT. 10) GO TO 503                DEF     212
         READ(UNITNO,1002) (CODE(I),VALUE(I),I=1,NOOFST)                DEF     213
         IF(EOF(UNITNO) .NE. 0.) GO TO 502                              DEF     214
         IF(ROCHK) GO TO 504                                            DEF     215
C                                                                       DEF     216
C     STORE CONVERSION VALUE INTO SURFACE TABLE                         DEF     217
C     SURFACE TABLE FORMAT                                              DEF     218
C                                                                       DEF     219
C     TABLE POSITION     ENTRY                                          DEF     220
C       SURFTB(I,1)      NUMERIC VALUE ASSOCIATED WITH 1                DEF     221
C          .                        .                                   DEF     222
C          .                        .                                   DEF     223
C          .                        .                                   DEF     224
C       SURFTB(I,10)     NUMERIC VALUE ASSOCIATED WITH 0                DEF     225
C                                                                       DEF     226
```

```
                C       SURFTB(I,11)     DIGIT IN V4 ASSIGNED TO THIS SURFACE
                C
           60   C     SPECIAL CODES
                C     OFF     999.
                C     NOTSET  (BLANK)
                C
                      DO 10 I=1,NOOFST
           65           ISUB=CODE(I)
                        IF(ISUB .LT. 0 .OR. ISUB .GT. 10) GO TO 504
                        IF(ISUB .EQ. 0) ISUB=10
                        SURFTB(DIGIT,ISUB)=VALUE(I)
                        GO TO 10
           70        10 CONTINUE
                     SURFTB(DIGIT,11)=DIGIT
                  20 CONTINUE
                     RETURN
                C
           75   C     TAPE CONVERSION ERRORS
                C
                501 CALL ERROR(25H(20H ERROR IN 9/C-NOOFDG),-3,ID)
                502 CALL ERROR(40H(35H UNEXPECTED END OF FILE ENCOUNTERED),-3,ID)
                503 CALL ERROR(27H(22H ERROR IN DIGIT-NOOFST),-3,ID)
           80   504 CALL ERROR(23H(18H ERROR IN SETTINGS),-3,ID)
               1000 FORMAT(F7.4,I3)
               1001 FORMAT(2I3)
               1002 FORMAT(11(I3,F7.2))
                     END
```

```
      OVERLAY(3,0)                                                      PARSE      2
      SUBROUTINE AXES, RETURNS(E)                                       PARSE      3
C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C* PARSE      4
C                                                                       PARSE      5
C        THIS SUBROUTINE SCANS THE REMAINDER OF THE INPUT RECORD AND     PARSE      6
C        BUILDS AN AXES BEAD IF THE FORMAT IS CORRECT AND THE ELEMENT    PARSE      7
C        NAMES ARE ALLOWABLE                                            PARSE      8
C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C* PARSE      9
      INTEGER XSTAT,YSTAT                                               PARSE     10
      DIMENSION IXWORD(20)                                              PARSE     11
      COMMON/CARDIN/IREC(80)                                            PARSE     12
      COMMON/SVECTOR/LPG,LAXIS,LPH,IESWT,LCONCRD                        PARSE     13
      COMMON/MLTAXES/ MLTAX                                             PARSE     14
      LOGICAL MLTAX                                                     PARSE     15
      COMMON/CCODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6    PARSE     16
      COMMON/LASTPOS/LPS                                                PARSE     17
      COMMON/ITABLE/IELETAB,IFUNTAB,XSCAL,YSCAL                         PARSE     18
      DATA ID /4HAXES/                                                  PARSE     19
C                                                                       PARSE     20
C        STAT  =  0  IMPLIES  NO STATUS                                 PARSE     21
C              =  1           AXIS=ELEMENT FOUND                       PARSE     22
C                                                                       PARSE     23
      NOFYAX=XSTAT=YSTAT=0                                              PARSE     24
      MLTAX = .FALSE.                                                   PARSE     25
C                                                                       PARSE     26
C        SCAN INPUT FOR X OR Y NAME                                     PARSE     27
   10 CALL SCAN(LPS,IREC,IWORD)                                         PARSE     28
      IF (IWORD .EQ. 0) GO TO 40                                        PARSE     29
C                                                                       PARSE     30
C        CHECK FOR X IN INPUT STREAM                                    PARSE     31
      IF (IWORD .NE. 10HX          ) GO TO 30                           PARSE     32
C                                                                       PARSE     33
C        GET X AXIS NAME                                                PARSE     34
      CALL SCAN(LPS,IREC,IWORD)                                         PARSE     35
      IF (XSTAT .EQ. 1) GO TO 9996                                      PARSE     36
      IF (IWORD .EQ. 0) GO TO 9995                                      PARSE     37
C                                                                       PARSE     38
C        CHECK IF ELEMENT NAME ACCEPTABLE                               PARSE     39
      CALL SEARCH(IELETAB, 1, IWORD)                                    PARSE     40
      IF (IWORD .EQ. 0) GO TO 9994                                      PARSE     41
      IXWORD = LOADVAL(IELETAB, 2, IWORD)                               PARSE     42
      XSTAT = 1                                                         PARSE     43
      GO TO 10                                                          PARSE     44
C                                                                       PARSE     45
C        CHECK FOR Y IN INPUT STREAM                                    PARSE     46
   30 IF (IWORD .NE. 10HY          ) GO TO 9991                         PARSE     47
      NOFYAX = NOFYAX + 1                                               PARSE     48
C                                                                       PARSE     49
C        GET Y AXIS NAME                                                PARSE     50
      CALL SCAN(LPS,IREC,IWORD)                                         PARSE     51
      IF(NOFYAX .LE. 20) GO TO 32                                       PARSE     52
      NOFYAX=20                                                         PARSE     53
      GO TO 9996                                                        PARSE     54
   32 CONTINUE                                                          PARSE     55
      IF (IWORD .EQ. 0) GO TO 9995                                      PARSE     56
C                                                                       PARSE     57
C        CHECK WHETHER ELEMENT NAME ACCEPTABLE                          PARSE     58
```

```
                  CALL SEARCH(IFLETAB, 1, IWORD)                                  PARSE      59
                  IF (IWORD .EQ. 0) GO TO 9994                                    PARSE      60
   60             IYWORD(NOFYAX) = LOADVAL(IELETAB, 2, IWORD)                     PARSE      61
                  YSTAT = 1                                                       PARSE      62
                  GO TO 10                                                        PARSE      63
               C                                                                  PARSE      64
               C     CHECK FOR MISSING X OR Y SPECIFICATION                       PARSE      65
   65          40 IF (XSTAT+YSTAT .NE. 2) GO TO 9997                             PARSE      66
               C                                                                  PARSE      67
               C     CHECK FOR DUPLICATE AXES NAMES                               PARSE      68
                  DO 50 I=1,NOFYAX                                                PARSE      69
                  IF (IXWORD .EQ. IYWORD(I)) GO TO 9992                           PARSE      70
   70          50 CONTINUE                                                       PARSE      71
                  IF (NOFYAX.EQ.1) GO TO 60                                       PARSE      72
                  DO 56 I=1,NOFYAX                                                PARSE      73
                  J=I                                                             PARSE      74
               55 CONTINUE                                                        PARSE      75
   75             J=J+1                                                          PARSE      76
                  IF(J .GT. NOFYAX) GO TO 56                                      PARSE      77
                  IF(IYWORD(I) .EQ. IYWORD(J)) GO TO 9992                         PARSE      78
                  GO TO 55                                                        PARSE      79
               56 CONTINUE                                                        PARSE      80
   80          C                                                                  PARSE      81
               C     CREATE AN AXIS BEAD                                          PARSE      82
               60 CALL DMGTBD(NOFYAX+6,IPTR)                                      PARSE      83
                  IF (IPTR .EQ. 0) GO TO 9993                                     PARSE      84
               C                                                                  PARSE      85
   85          C     STRING BEAD ONTO DATA STRUCTURE                              PARSE      86
                  IF (LAXIS .NE. 0) GO TO 70                                      PARSE      87
                  LAXIS = IPTR                                                    PARSE      88
                  CALL DMSET(IWHOLE4,LPG,IPTR)                                    PARSE      89
                  GO TO 80                                                        PARSE      90
   90             70 CONTINUE                                                     MODA       26
                  CALL DMSET(IWHOLE1,LAXIS,IPTR)                                  PARSE      94
                  LAXIS = IPTR                                                    PARSE      95
               C                                                                  PARSE      96
               C     FILL IN BEAD WITH AXES NAMES (NOW ARE VALUES)                PARSE      97
   95          80 CALL DMSET(IWHOLE2,LAXIS,IXWORD)                               PARSE      98
                  DO 90 I=1,NOFYAX                                                PARSE      99
                  CALL DMSET(IWHOLE4+I,LAXIS,IYWORD(I))                           PARSE     100
               90 CONTINUE                                                        PARSE     101
               C                                                                  PARSE     102
  100          C     PLACE NO OF Y AXES IN BEAD                                   PARSE     103
                  CALL DMSET(IWHOLE3,LAXIS,NOFYAX)                                PARSE     104
                  IF (NOFYAX .GT. 1) MLTAX=.TRUE.                                 PARSE     105
               C                                                                  PARSE     106
               C     SAVE TEXT AND SET SWITCHES                                   MODA       27
  105             CALL REFORM(NPTR,1)                                             MODA       28
                  CALL DMSET(IWHOLE5+NOFYAX,LAXIS,NPTR)                           MODA       29
                  LOP = 0                                                         PARSE     109
                  RETURN                                                          PARSE     110
               C                                                                  PARSE     111
  110          C     ERROR CALLS                                                  PARSE     112
               9991 CALL ERROR(24H(19H ARG MUST BE X OR Y),1,ID)                   PARSE     113
                  RETURN                                                          PARSE     114
               9992 CALL ERROR(42H(37H AXES X,Y ELEMENT NAMES ARE DUPLICATE),1,ID) PARSE     115
                  RETURN                                                          PARSE     116
```

```
115           9993 CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,ID)                    PARSE  117
              9994 CALL ERRORM(31H(26H ELEMENT NOT IN NAME TABLE),1,ID)                PARSE  118
                   RETURN E                                                            PARSE  119
              9995 CALL ERRORM(42H(37H ELEMENT NAME MISSING FOR X OR Y AXIS), 1,ID)    PARSE  120
                   RETURN E                                                            PARSE  121
120           9996 CALL ERRORM(36H(31H TOO MANY X OR Y AXES REQUESTED), 2,ID)          PARSE  122
                   GO TO 10                                                            PARSE  123
              9997 CALL ERRORM(39H(34H X OR Y AXIS SPECIFICATION MISSING), 1,ID)       PARSE  124
                   RETURN E                                                            PARSE  125
                   END                                                                 PARSE  126
```

09

```
      SUBROUTINE BLOPM                                                  PARSE    127
C                                                                       PARSE    128
C        THIS SUBROUTINE BUILDS AND ATTACHES A P-M BEAD FROM THE FOURTH PARSE    129
C        WORD OF THE LAXIS BEAD OR ONTO THE P-M LIST                    PARSE    130
C                                                                       PARSE    131
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONGRD                        PARSE    132
      COMMON/CCODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6    PARSE    133
C                                                                       PARSE    134
C        CREATE A P-M BEAD                                              PARSE    135
   10 IESWT = 0                                                         PARSE    136
      CALL LMGT3D( 7,IPTR)                                              PARSE    137
      IF (IPTR .EQ. 0) GO TO 9991                                       PARSE    138
C                                                                       PARSE    139
C        ATTACH P-M BEAD TO STRUCTURE                                   PARSE    140
   15 IF (LPM .NE. 0) GO TO 10                                          PARSE    141
      CALL DMSET(IWHOLE4,LAXIS,IPTR)                                    PARSE    142
      LPM = IPTR                                                        PARSE    143
      RETURN                                                            PARSE    144
   10 CALL DMSET(IWHOLE1,LPM,IPTR)                                      PARSE    145
   20 LPM = IPTR                                                        PARSE    146
      RETURN                                                            PARSE    147
 9991 CALL ERROFM(26H(21H BEAD SPACE EXHAUSTED),-3,SHBLOPM)             PARSE    148
      END                                                               PARSE    149
```

```
      SUBROUTINE CKLIST(ISTART,IMATCH,IEND,IND)                          PARSE   150
     C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*  PARSE   151
     C                                                                   PARSE   152
     C     THIS ROUTINE SEARCHES THROUGH A CONNECTED LIST OF THREE WORD BEADS PARSE   153
     C     LOOKING FOR A MATCH  WITH THE SECOND WORD OF THE BEAD         PARSE   154
     C     IF NONE IS FOUND A NEW BEAD IS CREATED, FILLED, AND ADDED TO THE PARSE   155
     C     DATA STRUCTURE                                                PARSE   156
     C        IN EITHER CASE A FUNCTION BEAD IS ATTACHED FROM THE THIRD WORD PARSE   157
     C        OF THE SELECTED BEAD                                       PARSE   158
     C        ISTART  - POINTER WHICH POINTS TO BEGINNING OF LIST (VALUE DEST PARSE   159
     C        IMATCH  - VALUE WHICH SHOULD BE MATCHED                    PARSE   160
     C        IEND    - PASSED IN , A POINTER TO A FUNCTION BEAD         PARSE   161
     C                - PASSED OUT, A POINTER TO THE SELECTED BEAD FROM  PARSE   162
     C                  WHICH THE FUNCTION BEAD WAS ATTACHED             PARSE   163
     C        IND     - LIST STATUS INDICATOR                            PARSE   164
     C                  VALUE OF 1 IMPLIES FIRST ADDITION TO LIST        PARSE   165
     C                           2         ADDITION TO LIST, NOT FIRST   PARSE   166
     C                           3         MATCH IN EXISTING STRUCTURE   PARSE   167
     C                                                                   PARSE   168
     C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*C*  PARSE   169
           COMMON/CCODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6 PARSE   170
           DATA ID /6HCKLIST/                                            PARSE   171
           IPTR = IEND                                                   PARSE   172
     C                                                                   PARSE   173
     C        CHECK IF LIST IS NULL                                      PARSE   174
           IF (ISTART .NE. 0) GO TO 35                                   PARSE   175
     C                                                                   PARSE   176
     C        CREATE A THREE WORD BEAD TO ADD TO THE DATA STRUCTURE      PARSE   177
           CALL CMGT3D( 3, ISTART)                                       PARSE   178
           IF (ISTART .EQ. 0) GO TO 9991                                 PARSE   179
           IND = 1                                                       PARSE   180
     C                                                                   PARSE   181
     C        STRING FUNCTION BEAD FROM THIRD WORD OF NEW BEAD           PARSE   182
        15 CALL CMSET(IWHOLE3,ISTART,IPTR)                               PARSE   183
     C                                                                   PARSE   184
     C        PLACE ELEMENT NAME IN SECOND WORD OF NEW BEAD              PARSE   185
           IEND = ISTART                                                 PARSE   186
           CALL CMSET(IWHOLE2,ISTART, IMATCH)                            PARSE   187
           RETURN                                                        PARSE   188
     C                                                                   PARSE   189
     C        SEARCH THROUGH BEAD LIST FOR MATCHING ELEMENT NAMES        PARSE   190
        35 IEND = ISTART                                                 PARSE   191
     C                                                                   PARSE   192
     C        GET THE DATA ELEMENT NAME FROM THE CURRENT BEAD            PARSE   193
           CALL CMGET(IWHOLE2,IEND,ICOMP)                                PARSE   194
     C                                                                   PARSE   195
     C        CHECK FOR MATCHING ELEMENT NAMES                           PARSE   196
           IF (IMATCH .NE. ICOMP) GO TO 45                               PARSE   197
     C                                                                   PARSE   198
     C        ELEMENT NAMES EQUAL                                        PARSE   199
           IND = 3                                                       PARSE   200
     C                                                                   PARSE   201
     C        DETERMINE WHERE TO STRING NEW FUNCTION BEAD                PARSE   202
           CALL CMGET(IWHOLE3,IEND,ISTART)                               PARSE   203
           IF (ISTART .EQ. 0) GO TO 9992                                 PARSE   204
        40 CALL CMGET(IWHOLE1,ISTART,IPT)                                PARSE   205
           IF (IPT .EQ. 0) GO TO 44                                      PARSE   206
```

```
                    ISTART = IPT                                                    PARSE    207
                    GO TO 40                                                        PARSE    208
60                44 CALL DMSET(IWHOLE1,ISTART,IPTR)                                PARSE    209
                    RETURN                                                          PARSE    210
                  45 CALL DMGET(IWHOLE1,IEND,ISTART)                                PARSE    211
                    IF (ISTART .EQ. 0) GO TO 55                                     PARSE    212
                    GO TO 35                                                        PARSE    213
65          C                                                                       PARSE    214
            C       CREATE A BEAD ADDITION TO LIST                                  PARSE    215
                  55 CALL DMGTBD( 3,ISTART)                                         PARSE    216
                    IF (ISTART .EQ. 0) GO TO 9991                                   PARSE    217
            C                                                                       PARSE    218
70          C       STRING NEW BEAD FROM LIST                                       PARSE    219
                    CALL DMSET(IWHOLE1,IEND,ISTART)                                 PARSE    220
                    IND = 2                                                         PARSE    221
                    GO TO 15                                                        PARSE    222
            C                                                                       PARSE    223
75          C       ERROR CALLS                                                     PARSE    224
             9991 CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,ID)                  PARSE    225
             9992 CALL ERRORM(32H(27H NO FUNCTION ATTCH TO EL BD),-3,ID)            PARSE    226
                    END                                                             PARSE    227
```

```
      SUBROUTINE COND                                                    PARSE    228
C                                                                        PARSE    229
C       THIS SUBROUTINE EVALUATES THE REMAINDER OF THE INPUT RECORD      PARSE    230
C       AND CORRECTLY APPENDS INFO TO THE DATA STRUCTURE IF VALID.       MODA      30
C                                                                        PARSE    234
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONCRD                         PARSE    235
      COMMON/CODES/IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6       PARSE    236
C                                                                        MODA      31
C       EVALUATE INPUT RECORD                                            MODA      32
      CALL DMGET(IWHOLE3,LPG,ISTART)                                     MODA      33
      CALL EVAL(ISTART,IPTR,NPTR,IND)                                    MODA      34
C                                                                        MODA      35
C       ATTACH INFO IF EVALUATION VALID                                  MODA      36
      IF (IND .EQ. 4) RETURN                                             MODA      37
      IF (IND .NE. 1) GO TO 15                                           PARSE    262
      CALL DMSET(IWHOLE3,LPG,IPTR)                                       PARSE    263
      CALL DMSET(IWHOLE5,LPG,NPTR)                                       PARSE    264
      CALL DMSET(IWHOLE6,LPG, 1)                                         PARSE    265
C                                                                        PARSE    266
C       INITIALIZE CONDITION TEXT READ POINTER                           PARSE    267
      LCONCRD = NPTR                                                     PARSE    268
      RETURN                                                             PARSE    269
C                                                                        PARSE    270
C       ATTACH PLOT DISPLAY READ AND FUNCTION READ TO THE CURRENT STRUC  PARSE    271
   15 CALL DMSET(IWHOLE1,LCONCRD,NPTR)                                   PARSE    272
      LCONCRD = NPTR                                                     PARSE    273
      CALL DMGET(IWHOLE6,LPG,NOFCON)                                     PARSE    274
C                                                                        PARSE    275
C       UPDATE THE CURRENT NUMBER OF CONDITIONS                          PARSE    276
      NOFCON = NOFCON + 1                                                PARSE    277
      CALL DMSET(IWHOLE6,LPG,NOFCON)                                     PARSE    278
      RETURN                                                             PARSE    279
      END                                                                PARSE    286
```

```
      SUBROUTINE DELLIST                                                PARSE    287
C        THIS SUBROUTINE DELETES THE LAST PLOT GROUP BEAD ALONG WITH     PARSE    288
C        ITS ASSOCIATED CONDITION AND FUNCTION LIST FROM THE DATA        PARSE    289
C        STRUCTURE                                                       PARSE    290
      COMMON/HEADPTR/IFPG                                               PARSE    291
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESHT,LCONCRD                        PARSE    292
      COMMON/CODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE    293
      IZERO = 0                                                         PARSE    294
      CALL ERROFM(44H(39H PREVIOUS PLOT GROUP INCOMPLETE,DELETED),3,    PARSE    295
     X 7HDELLIST)                                                       PARSE    296
C                                                                       PARSE    297
C        CHECK FOR NULL POINTER                                         PARSE    298
      IF (IFPG .EQ. 0) RETURN                                           PARSE    299
C                                                                       PARSE    300
C        ONLY ONE P-G BEAD                                              PARSE    301
      IF (IFPG .NE. LPG) GO TO 10                                       PARSE    302
      IFPG = IZERO                                                      PARSE    303
      GO TO 25                                                          PARSE    304
C                                                                       PARSE    305
C        SEARCH P-G LIST TO FIND NEXT TO LAST P-G BEAD                  PARSE    306
   10 IFPTR = IFPG                                                      PARSE    307
   15 CALL DMGET(IWHOLE1 ,IFPTR,ISPTR)                                  PARSE    308
      IF (ISPTR .EQ. LPG) GO TO 20                                      PARSE    309
      IFPTR = ISPTR                                                     PARSE    310
      GO TO 15                                                          PARSE    311
C                                                                       PARSE    312
C        ZERO OUT P-G POINTER IN NEXT TO LAST BEAD                      PARSE    3
   20 CALL DMSET(IWHOLE1,IFPTR,IZERO)                                   PARSE    3
C                                                                       PARSE    315
C        GET POINTER TO COND LIST FROM LAST BEAD                        PARSE    316
   25 CALL DMGET(IWHOLE3,LPG,ISPTR)                                     PARSE    317
C                                                                       PARSE    318
C        GET POINTERS TO TEXT BEADS                                     PARSE    319
      CALL DMGET(IWHOLE2,LPG,ITEXT1)                                    PARSE    320
      CALL DMGET(IWHOLE5,LPG,ITEXT2)                                    PARSE    321
C                                                                       PARSE    322
C        RELEASE THE LAST P-G BEAD AND UPDATE LPG                       PARSE    323
      CALL DMRLBD(LPG)                                                  PARSE    324
      LPG = IFPTR                                                       PARSE    325
C                                                                       PARSE    326
C        DELETE CONDITION AND FUNCTION LISTS, IF THEY ARE PRESENT       PARSE    327
   35 IF (ISPTR .EQ. 0) GO TO 60                                        PARSE    328
C                                                                       PARSE    329
C        GET POINTER TO FUNCTION BEAD TO BE DELETED                     PARSE    330
      CALL DMGET(IWHOLE3,ISPTR,IFPTR)                                   PARSE    331
C                                                                       PARSE    332
C        DELETE FUNCTION BEAD AND DETERMINE IF ANY MORE TO DELETE       PARSE    333
   40 CALL DMGET(IWHOLE1,IFPTR,INPTR)                                   PARSE    334
      CALL DMRLBD(IFPTR)                                                PARSE    335
      IF (INPTR .EQ. 0) GO TO 50                                        PARSE    336
      IFPTR = INPTR                                                     PARSE    337
      GO TO 40                                                          PARSE    338
C                                                                       PARSE    339
C        DELETE CONDITION BEAD                                          PARSE    340
   50 CALL DMGET(IWHOLE1,ISPTR,INPTR)                                   PARSE    341
      CALL DMRLBD(ISPTR)                                                PARSE    342
      ISPTR = INPTR                                                     PARSE    343
```

```
                GO TO 35                                                        PARSE     344
            C                                                                   PARSE     345
60          C       DELETE TITLE AND CONDITION TEXT BEADS                       PARSE     346
              60 IF (ITEXT1 .EQ. 0) GO TO 70                                    PARSE     347
            C                                                                   PARSE     348
            C       GET POINTER TO NEXT TEXT BEAD                               PARSE     349
                CALL CMGET(IWHOLE1,ITEXT1,INPTR)                                PARSE     350
65          C                                                                   PARSE     351
            C       DELETE TEXT BEAD                                            PARSE     352
                CALL CMRLBD(ITEXT1)                                             PARSE     353
                ITEXT1 = INPTR                                                  PARSE     354
                GO TO 60                                                        PARSE     355
70            70 IF (ITEXT2 .EQ. 0) RETURN                                      PARSE     356
            C                                                                   PARSE     357
            C       RESET POINTERS TO USE ABOVE CODE AGAIN FOR COND TEXT DELETION PARSE    358
                ITEXT1 = ITEXT2                                                 PARSE     359
                ITEXT2 = 0                                                      PARSE     360
75              GO TO 60                                                        PARSE     361
                END                                                             PARSE     362
```

```
      SUBROUTINE EVAL(ISTART,IPTR,NPTR,IND)                                  MODA   38
C                                                                            MODA   39
C        THIS SUBROUTINE EVALUATES THE REMAINDER OF THE INPUT RECORD         MODA   40
C        ATTEMPTING TO BUILD A FUNCTION BEAD.  IF ONE IS BUILT, IT           MODA   41
C        DETERMINES WHETHER TO EXPAND THE ELEMENT LIST FOR ITS               MODA   42
C        ATTACHMENT.                                                         MODA   43
C                                                                            MODA   44
C        ISTART - STARTING LOCATION IN ELEMENT LIST                          MODA   45
C        IPTR   - ELEMENT BEAD TO WHICH A FUNCTION BEAD HAS BEEN             MODA   46
C                 ATTACHED (RETURNED)                                        MODA   47
C        NPTR   - TEXT BEAD OF INPUT RECORD (RETURNED)                       MODA   48
C        IND    - ACTION INDICATOR                                           MODA   49
C                 (1 = FIRST ADDITION)  (2 = NEW ADDITION)                   MODA   50
C                 (3 = MATCH IN LIST)   (4 = ERROR SKIP)                     MODA   51
      COMMON  /LASTPOS/ LPS                                                  MODA   52
      COMMON  /CARDIN/ IREC(80)                                              MODA   53
      COMMON  /ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                           MODA   54
      DATA ID /4HEVAL/                                                       MODA   55
C                                                                            MODA   56
C        SEARCH INPUT STREAM FOR DELIMITED CHARACTERS                        MODA   57
      CALL SCAN(LPS,IREC,IWORD)                                              MODA   58
      IF (IWORD .EQ. 0) GO TO 9991                                           MODA   59
C                                                                            MODA   60
C        CHECK IF ELEMENT NAME ALLOWABLE                                     MODA   61
      CALL SEARCH(IELETAB, 1, IWORD)                                         MODA   62
      IF (IWORD .EQ. 0) GO TO 9992                                           MODA   63
C                                                                            MODA   64
C        ELEMENT NAME ALLOWABLE, CONTINUE EVALUATION                         MODA   65
      IELEN = LOADVAL(IELETAB,2, IWORD)                                      MODA   66
      ITOL = LOADVAL(IELETAB, 3, IWORD)                                      MODA   67
      CALL FUNCT(IPTR, ITOL)                                                 MODA   68
C                                                                            MODA   69
C        PLACE FUNCTION BEAD ON ELE LIST IF NO ERROR IN EVALUATION           MODA   70
      IF (IPTR .EQ. 0) GO TO 9999                                            MODA   71
      CALL CKLIST(ISTART,IELEN,IPTR,IND)                                     MODA   72
C                                                                            MODA   73
C        REFORMAT INPUT RECORD FOR DISPLAY PURPOSES                          MODA   74
      CALL REFORM(NPTR,1)                                                    MODA   75
      RETURN                                                                 MODA   76
C                                                                            MODA   77
C        ERROR CALLS                                                         MODA   78
 9991 CALL ERRORM(32H(27H NO DATA ELEMENT NAME GIVEN), 1, ID)                MODA   79
      GO TO 9999                                                             MODA   80
 9992 CALL ERRORM(31H(26H ELEMENT NOT IN NAME TABLE), 1, ID)                 MODA   81
 9999 IND = 4                                                                MODA   82
      RETURN                                                                 MODA   83
      END                                                                    MODA   84
```

```
      SUBROUTINE FUNCT(IPTR,ITOL)                                       PARSE   363
C                                                                       PARSE   364
C       IPTR   -  POINTER TO FUNCTION BEAD, IF ONE IS CREATED           PARSE   365
C              -  ZERO OTHERWISE                                        PARSE   366
C       ITOL   -  TABLE TOLERANCE                                       PARSE   367
C                                                                       PARSE   368
C       THIS SUBROUTINE SCANS AND EVALUATES THE REMAINDER OF THE INPUT  PARSE   369
C       RECORD AND ATTEMPTS TO BUILD REQUIRED FUNCTION BEAD AND RETURN  PARSE   370
C       THE POINTER TO IT                                               PARSE   371
C                                                                       PARSE   372
C       ISTAT  =  0  MEANS  LOOK FOR FIRST ARGUEMENT                    PARSE   373
C                 1         FIRST ARG FOUND, CONTINUE SEARCHING         PARSE   374
C                 2         FUNCTION ARGS SATISIFIED, LOOK FOR TOL      PARSE   375
C                 3         TOL FOUND                                   PARSE   376
C                 4         TOL=VALUE FOUND                             PARSE   377
C                                                                       PARSE   378
      COMMON/LASTPOS/LPS                                                PARSE   379
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                        PARSE   380
      COMMON/CARDIN/IREC(80)                                            PARSE   381
      COMMON/CODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE   382
      EQUIVALENCE (IVAL,VAL),(IZERO,ZERO)                               PARSE   383
      DATA ID /6HFUNCT/                                                 PARSE   384
      IPTR = 0                                                          PARSE   385
      ISTAT = 0                                                         PARSE   386
      ZERO = 0.                                                         PARSE   387
      KNOWALL = 0                                                       PARSE   388
C                                                                       PARSE   389
C       SCAN CARD FOR FUNCTION NAME                                     PARSE   390
      CALL SCAN(LPS,IREC,IWORD)                                         PARSE   391
      IF (IWORD .EQ. 0) GO TO 9991                                      PARSE   392
C                                                                       PARSE   393
C       CHECK FOR VALID FUNCTION NAME                                   PARSE   394
      CALL SEARCH(IFUNTAB, 1, IWORD)                                    PARSE   395
      IF(IWORD .EQ. 0) GO TO 9992                                       PARSE   396
C                                                                       PARSE   397
C       FUNCTION NAME CORRECT                                           PARSE   398
      IFN = LOADVAL(IFUNTAB, 2, IWORD)                                  PARSE   399
      IARG = LOADVAL(IFUNTAB, 3, IWORD)                                 PARSE   400
C                                                                       PARSE   401
C       SEARCH FOR IARG NUMBER OF ARGUEMENTS                            PARSE   402
      NEARG=0                                                           PARSE   403
      IFREAD = ILREAD = 0                                               PARSE   404
      DO 100 I=1,IARG                                                   PARSE   405
      CALL SCANE(LPS,IREC,VAL,ITYPE)                                    PARSE   406
C                                                                       PARSE   407
C          (EOS, VAL CONV, ERROR, ALL, TOL, OFF)                        PARSE   408
      GO TO(  10,    50   ,  20  ,  40,  30,  45),ITYPE                 PARSE   409
C                                                                       PARSE   410
C       END OF SCAN                                                     PARSE   411
   10 IF (ISTAT .NE. 1) GO TO 9993                                      PARSE   412
      ISTAT = 2                                                         PARSE   413
      IF (IARG .GE. 999) GO TO 130                                      PARSE   414
      GO TO 9993                                                        PARSE   415
C                                                                       PARSE   416
C       ERROR                                                           PARSE   417
   20 IF (IARG .GE. 999) GO TO 100                                      PARSE   418
      GO TO 9993                                                        PARSE   419
```

```
      C                                                                 PARSE     420
      C        TOL FOUND                                                PARSE     421
   60    30 IF (IARG .LT. 999) GO TO 9993                               PARSE     422
           IF (ISTAT .NE. 1) GO TO 9993                                 PARSE     423
           ISTAT = 3                                                    PARSE     424
           GO TO 110                                                    PARSE     425
      C                                                                 PARSE     426
   65 C        ALL FOUND IN SCAN                                        PARSE     427
      40 IF (ISTAT .NE. 0) GO TO 9993                                   PARSE     428
         NEARG = 1                                                      PARSE     429
         KNOWALL = 1                                                    PARSE     430
         ISTAT = 2                                                      PARSE     431
   70    IF (IARG .GE. 999) GO TO 110                                   PARSE     432
         GO TO 9996                                                     PARSE     433
      C                                                                 PARSE     434
      C        OFF FOUND IN SCAN                                        PARSE     435
      45 VAL = 999.                                                     PARSE     436
   75 C                                                                 PARSE     437
      C        VALUE CONVERTED                                          PARSE     438
      50 IF (ISTAT .NE. 0 .AND. ISTAT .NE. 1) GO TO 9993                PARSE     439
         ISTAT = 1                                                      PARSE     440
         NEARG = NEARG + 1                                              PARSE     441
   80    IF (NEARG .EQ. IARG) ISTAT = 2                                 PARSE     442
      C                                                                 PARSE     443
   69 C        CREATE AND FILL AN ARGUEMENT BEAD                        PARSE     444
         CALL DMGT80(2,IPT)                                             PARSE     445
         IF (IPT .EQ. 0) GO TO 9994                                     PARSE     446
   85    CALL DMSET(IWHOLE2,IPT,VAL)                                    PARSE     447
         IF (ILBEAD .NE. 0) GO TO 55                                    PARSE     448
         IFBEAD = ILBEAD = IPT                                          PARSE     449
         GO TO 100                                                      PARSE     450
      55 CALL DMSET(IWHOLE1,ILBEAD,IPT)                                 PARSE     451
   90    ILBEAD = IPT                                                   PARSE     452
     100 CONTINUE                                                       PARSE     453
      C                                                                 PARSE     454
      C        AT LEAST ALL FUNCTION ARGS ARE SATISFIED (ISTAT EQUALS 2 OR 3)  PARSE  455
         IF (ISTAT .NE. 2 .AND. ISTAT .NE. 3) GO TO 9995                PARSE     456
   95  110 CALL SCANE(LFS,IREC,VAL,ITYPE)                               PARSE     457
      C                                                                 PARSE     458
      C        LOOK FOR TOL                                             PARSE     459
         IF (ISTAT .NE. 2) GO TO 120                                    PARSE     460
         IF (ITYPE .EQ. 1) GO TO 130                                    PARSE     461
  100    IF (ITYPE .NE. 5) GO TO 9993                                   PARSE     462
         IF (KNOWALL.EQ. 1) GO TO 9997                                  PARSE     463
         ISTAT = 3                                                      PARSE     464
         GO TO 110                                                      PARSE     465
      C                                                                 PARSE     466
  105 C        LOOK FOR VALUE                                           PARSE     467
     120 GO TO (130,125,130,9993,9993,9993),ITYPE                       PARSE     468
     125 ITOL = IVAL                                                    PARSE     469
         ISTAT = 4                                                      PARSE     470
     130 IF (ISTAT .EQ. 2) ITOL = IZERO                                 PARSE     471
  110 C                                                                 PARSE     472
      C        CREATE A FUNCTION BEAD                                   PARSE     473
         IF (IARG .GE. 999) IARG = NEARG                                PARSE     474
         IF (KNOWALL .EQ. 1) IARG = IARG -1                             PARSE     475
         N = IARG +4                                                    PARSE     476
```

```
      CALL DMGTBD(N,IPTR)                                                PARSE   477
      IF (IPTR .EQ. 0) GO TO 9994                                        PARSE   478
C                                                                        PARSE   479
C        FILL IN BEAD WITH STANDARD VALUES                               PARSE   480
      CALL DMSET(IWHOLE2,IPTR,IFN)                                       PARSE   481
      CALL DMSET(IWHOLE3,IPTR,IARG)                                      PARSE   482
      CALL DMSET(IWHOLE4,IPTR,ITOL)                                      PARSE   483
C                                                                        PARSE   484
C        FILL IN REMAINDER OF BEAD WITH FUNCTION ARGUEMENTS              PARSE   485
      IWHOLEN = 9100000000005B                                           PARSE   486
  150 IF (IARG .EQ. 0) RETURN                                            PARSE   487
      CALL DMGET(IWHOLE2,IFBEAD,VAL)                                     PARSE   488
      CALL DMGET(IWHOLE1,IFBEAD,INBEAD)                                  PARSE   489
      CALL DMPLOD(IFBEAD)                                                PARSE   490
      CALL DMSET(IWHOLEN,IPTR,VAL)                                       PARSE   491
      IFBEAD=INBEAD                                                      PARSE   492
      IARG = IARG -1                                                     PARSE   493
      IWHOLEN = IWHOLEN + 1B                                             PARSE   494
      GO TO 150                                                          PARSE   495
C                                                                        PARSE   496
C        ERROR CALLS                                                     PARSE   497
 9991 CALL ERRORM(28H(23H NO FUNCTION NAME GIVEN),1,ID)                  PARSE   498
      RETURN                                                             PARSE   499
 9992 CALL ERRORM(37H(32H FUNCTION NAME NOT IN NAME TABLE),1,ID)         PARSE   500
      RETURN                                                             PARSE   501
 9993 CALL ERRORM(37H(32H ERROR IN THE FUNCTION ARGUMENTS),1,ID)         PARSE   502
      RETURN                                                             PARSE   503
 9994 CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,ID)                   PARSE   504
 9995 CALL ERRORM(26H(21H INTERNAL LOGIC ERROR),-3,ID)                   PARSE   505
 9996 CALL ERRORM(41H(36H FUNCTION NOT PERMITTED ALL ARGUMENT),1,ID)     PARSE   506
      RETURN                                                             PARSE   507
 9997 CALL ERRORM(37H(32H ALL ARG NOT PERMITTED TOL VALUE),1,ID)         PARSE   508
      RETURN                                                             PARSE   509
      END                                                                PARSE   510
```

```
      SUBROUTINE LABEL                                                  PARSE   511
C                                                                       PARSE   512
C        THIS SUBROUTINE EVALUATES THE REMAINDER OF THE CURRENT INPUT   PARSE   513
C        RECORD, SETTING UP A LABEL REQUEST FOR EACH ELEMENT NAME IT    PARSE   514
C        ENCOUNTERS IN THE SCAN.                                        PARSE   515
C                                                                       PARSE   516
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                        PARSE   517
      COMMON/SVECTOR/ LPG,LAXIS,LPM, IESWT,LCONCRD                      PARSE   518
      COMMON/LASTPOS/ LPS                                               PARSE   519
      COMMON/CARDIN/ IREC(80)                                           PARSE   520
      COMMON/CCODES/IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE   521
      LBLPTR = 0                                                        PARSE   522
      LBLCNT = 0                                                        PARSE   523
C                                                                       PARSE   524
C        SEARCH INPUT RECORD FOR ELEMENT NAME                           PARSE   525
    5 CALL SCAN(LPS, IREC, IWORD)                                       PARSE   526
      IF (IWORD .EQ. 0) RETURN                                          PARSE   527
C                                                                       PARSE   528
C        ELEMENT NAME ACCEPTABLE                                        PARSE   529
      CALL SEARCH(IELETAB,1,IWORD)                                      PARSE   530
      IF (IWORD .EQ. 0) GO TO 9991                                      PARSE   531
      IELEN = LOADVAL(IELETAB, 2,IWORD)                                 PARSE   532
C                                                                       PARSE   533
C        CREATE AND FILL LABEL BEAD                                     PARSE   534
      CALL DMGTBD(3,IPTR)                                               PARSE   535
      IF (IPTR .EQ. 0) GO TO 9992                                       PARSE   536
      CALL DMSET(IWHOLE2,IPTR,IELEN)                                    PARSE   537
C                                                                       PARSE   538
C        LINK LBL BEAD TO PREV. - OR PG BEAD                            PARSE   539
      IF (LBLPTR .NE. 0) GO TO 10                                       PARSE   540
      CALL DMSET(IWHOLE6+1,LPG,IPTR)                                    PARSE   541
      GO TO 20                                                          PARSE   542
   10 CALL DMSET(IWHOLE1,LBLPTR,IPTR)                                   PARSE   543
   20 LBLPTR = IPTR                                                     PARSE   544
C                                                                       PARSE   545
C        ADJUST LABEL COUNT                                             PARSE   546
      LBLCNT = LBLCNT + 1                                               PARSE   547
      CALL DMSET(IWHOLE2+6,LPG,LBLCNT)                                  PARSE   548
      GO TO 5                                                           PARSE   549
C                                                                       PARSE   550
C        ERROR CALLS                                                    PARSE   551
 9991 CALL ERRORM(31H(26H ELEMENT NOT IN NAME TABLE), 2,ID)             PARSE   552
      GO TO 5                                                           PARSE   553
 9992 CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,ID)                  PARSE   554
      END                                                               PARSE   555
```

```
      INTEGER FUNCTION LOADVAL(ITABPTR,ITABCOL,ITABROW)                 PARSE   556
      COMMON/CCODES/IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE   557
      DATA IWHOLE0/01000000000B/                                        PARSE   558
C                                                                       PARSE   559
C     GET POINTER TO CORRECT TABLE COL                                  PARSE   560
      CALL DMGET(IWHOLE2+ITABCOL,ITABPTR,ICOLPTR)                       PARSE   561
C                                                                       PARSE   562
C     GET VALUE FROM SELECTED ROW                                       PARSE   563
      CALL DMGET(IWHOLE0+ITABROW,ICOLPTR,LOADVAL)                       PARSE   564
      RETURN                                                            PARSE   565
      END                                                               PARSE   566
```

```
      SUBROUTINE MULT                                                    PARSE     567
C                                                                        PARSE     568
C        THIS SUBROUTINE EVALUATES THE REMAINDER OF THE INPUT RECORD,    PARSE     569
C        AND IF VALID ATTACHES THE RELATED INFO TO THE DATA STRUCTURE.   MODA       85
C                                                                        PARSE     572
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONCRD                         PARSE     573
      COMMON/CCODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE     574
C                                                                        MODA       86
C        EVALUATE INPUT RECORD                                           MODA       87
      CALL DMGET(IWHOLE3,LPM,ISTART)                                     MODA       88
      CALL EVAL(ISTART,IPTR,NPTR,IND)                                    MODA       89
C                                                                        MODA       90
C        ATTACH RELATED INFO IF VALID                                    MODA       91
      GO TO (10, 20, 20, 40), IND                                        MODA       92
   10 CALL DMSET(IWHOLE3,LPM,IPTR)                                       MODA       93
      CALL DMSET(IWHOLE1+6,LPM,NPTR)                                     MODA       94
      LMCRD = NPTR                                                       MODA       95
      CALL DMSET(IWHOLE4,LPM, 1)                                         MODA       96
      GO TO 40                                                           MODA       97
   20 CALL DMSET(IWHOLE1,LMCRD,NPTR)                                     MODA       98
      LMCRD = NPTR                                                       MODA       99
      CALL DMGET(IWHOLE4,LPM,NMULTS)                                     MODA      100
      IF (IND .NE. 3) NMULTS = NMULTS + 1                                MODA      101
      CALL DMSET(IWHOLE4,LPM,NMULTS)                                     MODA      102
   40 CONTINUE                                                           MODA      103
      RETURN                                                             PARSE     614
      END                                                                PARSE     615
```

```
      SUBROUTINE PARA                                                   PARSE   616
C                                                                       PARSE   617
C     THIS SUBROUTINE EVALUATES THE REMAINDER OF THE INPUT RECORD,      PARSE   618
C        AND IF VALID ATTACHES RELATED INFO TO THE CURRENT STRUCTURE.   MODA    104
C                                                                       PARSE   621
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONCRD                        PARSE   622
      COMMON/CCODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6    PARSE   623
      COMMON/MLTAXES/ MLTAX                                             PARSE   627
      LOGICAL MLTAX                                                     PARSE   628
      DATA ID /4HPARA/                                                  PARSE   629
C                                                                       PARSE   630
C     CHECK IF PARAMETER NOT ALLOWED - MULTIPLE AXES SITUATION          PARSE   631
C                                                                       PARSE   632
      IF (MLTAX) GO TO 9992                                             PARSE   633
C                                                                       MODA    105
C        EVALUATE INPUT RECORD                                          MODA    106
      CALL DMGET(IWHOLE2,LPM,ISTART)                                    MODA    107
      CALL EVAL(ISTART,IPTR,NPTR,IND)                                   MODA    108
C                                                                       MODA    109
C        ATTACH RELATED INFO IF VALID                                   MODA    110
      GO TO (10, 20, 20, 40), IND                                       MODA    111
   10 CALL DMSET(IWHOLE2,LPM,IPTR)                                      MODA    112
      CALL DMSET(IWHOLE6,LPM,NPTR)                                      MODA    113
      LPCRD = NPTR                                                      MODA    114
      GO TO 40                                                          MODA    115
   20 CALL DMSET(IWHOLE1,LPCRD,NPTR)                                    MODA    116
      LPCRD = NPTR                                                      MODA    117
   40 CONTINUE                                                          MODA    118
      RETURN                                                            MODA    119
 9992 CALL ERRORM(61H(56H PARAMETERS NOT ALLOWED WITH MULTIPLE AXES SPEC PARSE   660
     XIFICATION), 1, ID)                                                PARSE   661
      RETURN                                                            PARSE   662
      END                                                               PARSE   665
```

```
      SUBROUTINE PLOT1, RETURNS(FINISH)                                  PARSE    666
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-* PARSE    667
C                                                                        PARSE    668
C        THIS SUBROUTINE CREATES A PLOT GROUP BEAD AND CORRECTLY JOINS   PARSE    669
C        IT INTO THE EXISTING DATA STRUCTURE                             PARSE    670
C        LPS     -  LAST POSITION SCANNED                                PARSE    671
C                                                                        PARSE    672
C        * PLOT GROUP BEAD STRUCTURE *                                   PARSE    673
C                 *-*-*-*-*-*-*-*-*-*                                    PARSE    674
C         WORD 1  * PTR TO NEXT BEAD  *                                  PARSE    675
C         WORD 2  * PTR TO TITLE BEAD *                                  PARSE    676
C         WORD 3  * PTR TO ELE LIST   *                                  PARSE    677
C         WORD 4  * PTR TO AXES LIST  *                                  PARSE    678
C         WORD 5  * PTR TO COND TEXT  *                                  PARSE    679
C         WORD 6  * NUM OF COND BEADS *                                  PARSE    680
C        WORD 7  * PTR TO LABEL LIST *                                   PARSE    681
C        WORD 8  * NUM OF LBL BEADS  *                                   PARSE    682
C                 *-*-*-*-*-*-*-*-*-*                                    PARSE    683
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-* PARSE    684
      COMMON/HEADPTR/IFPG                                                PARSE    685
      COMMON/CARDIN/IREC(80)                                             PARSE    686
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONCRD                         PARSE    687
      COMMON/CCODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE    688
C                                                                        PARSE    689
C        CREATE AN EIGHT WORD PG BEAD                                    PARSE    690
      CALL CMGTBD(9,IPTR)                                                PARSE    691
      IF (IPTR .EQ. 0) GO TO 999                                         PARSE    692
      IF (IFPG .NE. 0) GO TO 15                                          PARSE    693
C                                                                        PARSE    694
C        INITIALIZE DATA STRUCTURE POINTER AND STATUS VECTOR ELEMENT     PARSE    695
      IFPG = IPTR                                                        PARSE    696
      LPG  = IPTR                                                        PARSE    697
      GO TO 99999                                                        PARSE    698
C                                                                        PARSE    699
C        STRING ON NEW BEAD AND UPDATE STATUS VECTOR                     PARSE    700
   15 CALL CMSET(IWHOLE1,LPG,IPTR)                                       PARSE    701
      LPG = IPTR                                                         PARSE    702
C                                                                        PARSE    703
C        CREATE AN AXES BEAD IF NECESSARY                                PARSE    704
      IF(IESWT .EQ. 1) CALL BLDPM                                        PARSE    705
C                                                                        PARSE    706
C        INITIALIZE PORTIONS OF STATUS VECTOR                            PARSE    707
99999 CONTINUE                                                           NOCA     120
      LAXIS = 0                                                          PARSE    709
      LPM   = 0                                                          PARSE    710
      RETURN                                                             PARSE    711
C                                                                        PARSE    712
C        ERROR ROUTINE                                                   PARSE    713
  999 CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,4HPLOT)               PARSE    714
      END                                                                PARSE    715
```

```
      PROGRAM PS2MAIN                                                   PARSE    716
C                                                                       PARSE    717
C       THIS PROGRAM IN ADDITION TO PERFORMING THE INITIALIZATION AND   PARSE    718
C       DRIVING THE REMAINING SUBPROGRAMS, CHECKS THE KEYWORDS ON THE   PARSE    719
C       INCOMING RECORDS FOR VALIDITY AND PROPER SEQUENCING             PARSE    720
C                                                                       PARSE    721
      COMMON/HF/OPTB/IFPG                                               PARSE    722
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONCRD                        PARSE    723
      COMMON/LASTPOS/LPS                                                PARSE    724
      COMMON/CCODES/IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE    725
      COMMON/CARDIN/IREC(80)                                            PARSE    726
      COMMON /ITABLES/ IELETAB(90,3),IELEROW,IELECOL,NOOFEL,            PARSE    727
     X IFUNTAB(10,3),IFUNROW,IFUNCOL,NOOFFN,ISURFTB(10,11)              PARSE    728
      COMMON/IOUNIT/IN,IO,IA1,IO1,IN2,IO2,IN3,IO3                       PARSE    729
      COMMON/ERROR/CHECK                                                PARSE    730
      COMMON/MLTAXES/ MLTAX                                             PARSE    731
      LOGICAL MLTAX                                                     PARSE    732
      INTEGER ZTRUNC                                                    PARSE    733
      LOGICAL CHECK,PLOTSWT,AERSWT                                      PARSE    734
      DATA PLOTSWT,AERSWT/.TRUE.,.FALSE./                               PARSE    735
      DATA IP,IL,IC,IA,IM,IE,JF/1LP,1LL,1LC,1LA,1LM,1LE,1LF/            PARSE    736
      DATA ID /7HPS2MAIN/                                               PARSE    737
C ///// REMOVE /////     //////                                         PARSE    738
      WRITE(IO1,1)                                                      PARSE    739
    1 FORMAT(1H1,/////,20X,19HPLOT SPECIFICATIONS,///)                  PARSE    740
C                                                                       PARSE    741
C                                                                       PARSE    742
C       READ IN INPUT RECORD                                            PARSE    743
    5 READ(IN1,10) IREC                                                 PARSE    744
   10 FORMAT(80A1)                                                      PARSE    745
      IF(ECF(IN1)) 90,11                                                PARSE    746
   11 WRITE(IO1,13) IREC                                                PARSE    747
   13 FORMAT(1H ,80A1)                                                  PARSE    748
      LPS = 1                                                           PARSE    749
C                                                                       PARSE    750
C       SEARCH INPUT RECORD FOR KEYWORD                                 PARSE    751
      CALL SCAN(LPS,IREC,IWORD)                                         PARSE    752
      IF (IWORD .EQ. 0) GO TO 9991                                      PARSE    753
      IF (ZTRUNC(IWORD,1) .EQ. JF) GO TO 90                             PARSE    754
      ICK = ZTRUNC(IWORD,2)                                             PARSE    755
C                                                                       PARSE    756
C       CHECK KEYWORD TYPE                                              PARSE    757
C                                                                       PARSE    758
C                                                                       PARSE    759
C       PLOT GROUP CARD                                                 PARSE    760
   20 IF (ICK .NE. IL) GO TO 30                                         PARSE    761
      IF (PLOTSWT) GO TO 25                                             PARSE    762
      IF (LAXIS .EQ. 0 .AND. .NOT.AERSWT) GO TO 9992                    PARSE    763
      IF (.NOT. AERSWT) GO TO 25                                        PARSE    764
      AERSWT = .FALSE.                                                  PARSE    765
      IF (LAXIS .EQ. 0) CALL DELLIST                                    PARSE    766
   25 PLOTSWT = .FALSE.                                                 PARSE    767
      CALL PLOT1, RETURNS(90)                                           PARSE    768
      GO TO 5                                                           PARSE    769
C                                                                       PARSE    770
C       AXES CARD                                                       PARSE    771
   30 ICK = ZTRUNC(IWORD,1)                                             PARSE    772
```

```
                IF (ICK .NE. IA) GO TO 40                                  PARSE   773
                IF (LFG .EQ. 0) GO TO 9993                                 PARSE   774
60              IF (AERSWT) AERSWT = .FALSE.                               PARSE   775
                CALL AXES, RETURNS(35)                                     PARSE   776
                CALL BLOPM                                                 MODA    121
                GO TO 5                                                    PARSE   777
          C                                                                PARSE   778
65        C     ERROR ON AXES CARD                                         PARSE   779
             35 AERSWT = .TRUE.                                            PARSE   780
                GO TO 5                                                    PARSE   781
          C                                                                PARSE   782
          C     DETERMINE WHETHER TO SKIP CARDS                            PARSE   783
70           40 IF (AERSWT) GO TO 99990                                    PARSE   784
          C                                                                PARSE   785
          C     CONDITION CARD                                             PARSE   786
                IF (ICK .NE. IC) GO TO 45                                  PARSE   787
                IF (PLOTSWT) GO TO 9994                                    PARSE   788
75              IF (LAXIS .NE. 0) GO TO 9994                               PARSE   789
                CALL COND                                                  PARSE   790
                GO TO 5                                                    PARSE   791
          C                                                                PARSE   792
          C     LABEL REQUEST CARD                                         PARSE   793
80           45 IF (ICK .NE. IL) GO TO 50                                  PARSE   794
                IF (PLOTSWT) GO TO 99992                                   PARSE   795
                IF (LAXIS .NE. 0) GO TO 99992                              PARSE   796
                CALL LABEL                                                 PARSE   797
                GO TO 5                                                    PARSE   798
85        C                                                                PARSE   799
          C     TITLE CARD                                                 PARSE   800
             50 IF (ICK .NE. ILT) GO TO 55                                 PARSE   801
                IF (PLOTSWT) GO TO 99991                                   PARSE   802
                IF(LAXIS .NE. 0) GO TO 99991                               PARSE   803
90              CALL TITLE                                                 PARSE   804
                GO TO 5                                                    PARSE   805
          C                                                                PARSE   806
          C     PARAMETER CARD                                             PARSE   807
             55 IF (ICK .NE. IP) GO TO 60                                  PARSE   808
95              IF(PLOTSWT) GO TO 9995                                     PARSE   809
                IF (LAXIS .EQ. 0) GO TO 9995                               PARSE   810
                IF (IESWT .EQ. 1) CALL BLOPM                               MODA    122
                CALL PARA                                                  PARSE   811
                GO TO 5                                                    PARSE   812
100       C                                                                PARSE   813
          C     MULTIPLIER CARD                                            PARSE   814
             60 IF (ICK .NE. IM) GO TO 70                                  PARSE   815
                IF (PLOTSWT) GO TO 9996                                    PARSE   816
                IF (LAXIS .EQ. 0) GO TO 9996                               PARSE   817
105             IF (IESWT .EQ. 1) CALL BLOPM                               MODA    123
                CALL MULT                                                  PARSE   818
                GO TO 5                                                    PARSE   819
          C                                                                PARSE   820
          C     END CARD                                                   PARSE   821
110          70 IF (ICK .NE. IE) GO TO 80                                  PARSE   822
                IF(PLOTSWT) GO TO 9997                                     PARSE   823
                IF (LAXIS .EQ. 0) GO TO 9997                               PARSE   824
                IESWT = 1                                                  MODA    124
                GO TO 5                                                    MODA    125
```

```
      C                                                                  PARSE     855
      C        FINISH CARD                                               PARSE     856
         80 IF (ICK .NE. JF) GO TO 9999                                  PARSE     857
         90 IF (PLOTSWT) GO TO 9998                                      PARSE     858
            IF (LAXIS .EQ. 0 .AND. .NOT.AERSWT) GO TO 9998               PARSE     859
         99 IF(LAXIS .EQ. 0) CALL DELLIST                                PARSE     860
            IF (IESWT .EQ. 1 .AND. LAXIS .NE. 0) CALL BLDPM              PARSE     861
            GO TO 99999                                                  PARSE     862
      C                                                                  PARSE     863
      C        ERROR CALLS                                               PARSE     864
       9991 CALL ERRORM(36H(31H NO INFORMATION ON INPUT RECORD),1,IO)     PARSE     865
            GO TO 5                                                      PARSE     866
       9992 CALL ERRORM(44H(39H IMPROPER OCCURRENCE OF PLOT GROUP CARD),1,IO)  PARSE  867
            GO TO 5                                                      PARSE     868
       9993 CALL ERRORM(39H(33H IMPROPER OCCURRENCE OF AXES CARD),1,IO)   PARSE     869
            GO TO 5                                                      PARSE     870
       9994 CALL ERRORM(47H(38H IMPROPER OCCURRENCE OF CONDITION CARD),1,IO)  PARSE  871
            GO TO 5                                                      PARSE     872
       9995 CALL ERRORM(43H(36H IMPROPER OCCURRENCE OF PARAMETER CARD),1,IO)  PARSE  873
            GO TO 5                                                      PARSE     874
       9996 CALL ERRORM(44H(39H IMPROPER OCCURRENCE OF MULTIPLIER CARD),1,IO)  PARSE  875
            GO TO 5                                                      PARSE     876
       9997 CALL ERRORM(37H(32H IMPROPER OCCURRENCE OF END CARD),1,IO)     PARSE     877
            GO TO 5                                                      PARSE     878
       9998 CALL ERRORM(47H(42H IMPROPER OCCURRENCE OF FINISH CARD OR EOF), 3,  PARSE  879
           X IO)                                                         PARSE     880
            GO TO 99                                                     PARSE     881
       9999 CALL ERRORM(28H(23H UNRECOGNIZABLE KEYWORD),1,IO)             PARSE     882
            GO TO 5                                                      PARSE     883
      99990 CALL ERRORM(41H(36H INPUT NOT A PLOT GROUP OR AXES CARD),1,IO)  PARSE   884
            GO TO 5                                                      PARSE     885
      99991 CALL ERRORM(39H(34H IMPROPER OCCURRENCE OF TITLE CARD),1,IO)  PARSE     886
            GO TO 5                                                      PARSE     887
      99992 CALL ERRORM(39H(34H IMPROPER OCCURRENCE OF LABEL CARD), 1,IO) PARSE     888
            GO TO 5                                                      PARSE     889
      99999 CONTINUE                                                     PARSE     890
            END                                                          PARSE     891
```

```
      SUBROUTINE REFORM(IPTR,IND)                                        PARSE   892
C                                                                        PARSE   893
C        THIS SUBROUTINE REFORMATS THE CURRENT INPUT RECORD AND DELETES  PARSE   894
C        UNNECESSARY BLANKS ALONG WITH THE KEYWORD IF IND=1.             PARSE   895
C        A TEN WORD TEXT BEAD IS CREATED AND FILLED WITH THE INPUT RECOR PARSE   896
C        AND THE POINTER TO IT IS RETURNED.                              PARSE   897
C                                                                        PARSE   898
      COMMON/ CODES/ IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE   899
      COMMON/CARDIN/IREC(80)                                             PARSE   900
      COMMON/LASTPOS/LPS                                                 PARSE   901
      DIMENSION IHOLD(8)                                                 PARSE   902
C                                                                        PARSE   903
C        REMOVE UNNECESSARY BLANKS                                       PARSE   904
      IF (IND .EQ. 1) CALL SQUEEZE(IREC)                                 PARSE   905
C                                                                        PARSE   906
C        FIND THE FIRST NON-BLANK CHARACTER IN TEXT CARD                 PARSE   907
      L = 1                                                              PARSE   908
      IF (IND .EQ. 2) L = LPS                                            PARSE   909
      DO 55 J=L,80                                                       PARSE   910
      IF (IREC(J) .NE. 1H ) GO TO 60                                     PARSE   911
   55 CONTINUE                                                           PARSE   912
C                                                                        PARSE   913
C        BLANK CARD                                                      PARSE   914
      NCHAR = 0                                                          PARSE   915
      GO TO 80                                                           PARSE   916
C                                                                        PARSE   917
C        FIND THE LAST NON-BLANK CHARACTER                               PARSE   [illegible]
   60 DO 65 K=J,80                                                       PARSE   [illegible]
      L = 80 -K +J                                                       PARSE   920
      IF (IREC(L) .NE. 1H ) GO TO 70                                     PARSE   921
   65 CONTINUE                                                           PARSE   922
      L = J                                                              PARSE   923
C                                                                        PARSE   924
C        DETERMINE THE NUMBER OF CHARACTERS                              PARSE   925
   70 NCHAR = L - J +1                                                   PARSE   926
C                                                                        PARSE   927
C        BLANK OUT IHOLD                                                 PARSE   928
      DO 75 K = 1,8                                                      PARSE   929
   75 IHOLD(K) = 10H                                                     PARSE   930
C                                                                        PARSE   931
C        ENCODE NCHAR CHARACTERS OF IREC INTO IHOLD                      PARSE   932
      ENCODE(10,77,IHOLD) (IREC(K),K=J,L)                                PARSE   933
   77 FORMAT(10A1)                                                       PARSE   934
C                                                                        PARSE   935
C        GET A 10 WORD PG TEXT BEAD                                      PARSE   936
   80 CALL PMGTBD(10,IPTR)                                               PARSE   937
      IF (IPTR .EQ. 0) GO TO 9991                                        PARSE   938
C                                                                        PARSE   939
C        FILL TEXT BEAD                                                  PARSE   940
      CALL PMSET(IWHOLE2,IPTR,NCHAR)                                     PARSE   941
      CALL PMSET(0700100000033,IPTR,IHOLD)                               PARSE   942
      RETURN                                                             PARSE   943
C                                                                        PARSE   944
C        ERROR CALLS                                                     PARSE   945
 9991 CALL ERROPM(26H(21H BEAD SPACE EXHAUSTED),-3,6HREFORM)             PARSE   946
      END                                                                PARSE   947
```

```
      SUBROUTINE SCAN(IFROM,ARRAY,IWORD)                                PARSE     948
     C-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*- PARSE     949
     C                                                                  PARSE     950
     C       THIS SUBROUTINE SCANS AN INPUT ARRAY FOR A FIELD OF CONTINUOUS PARSE     951
     C       CHARACTERS AND RETURNS THE LEFTMOST TEN LEFT JUSTIFIED AND BLAN PARSE     952
     C       IFROM   -  THE POINT WHERE SCANNING BEGINS, PASSED IN      PARSE     953
     C               -  ONE PAST THE LAST POSITION SCANNED, PASSED OUT  PARSE     954
     C       ARRAY   -  ARRAY INTO WHICH THE INPUT RECORD HAS BEEN READ UNDE PARSE     955
     C       IWORD   -  WORD CONTAINING SCANNED CHARACTERS              PARSE     956
     C-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*- PARSE     957
           INTEGER ARRAY(1)                                             PARSE     958
           DATA IBLK/10H                                                PARSE     959
           DATA ICOMMA/10H,                                             PARSE     960
           DATA IEQ/10H=                                                PARSE     961
           IF (IFROM .GT. 80) GO TO 15                                  PARSE     962
     C                                                                  PARSE     963
     C                                                                  PARSE     964
           DO 10 J=IFROM,80                                             PARSE     965
           IF (ARRAY(J) .NE. IBLK .AND. ARRAY(J) .NE. ICOMMA            PARSE     966
          X .AND. ARRAY(J) .NE. IEQ) GO TO 20                           PARSE     967
        10 CONTINUE                                                     PARSE     968
     C                                                                  PARSE     969
     C       IF IWORD EQUALS ZERO THEN NO CHARACTERS REMAIN             PARSE     970
        15 IWORD = 0                                                    PARSE     971
           RETURN                                                       PARSE     972
     C                                                                  PARSE     973
     C       FIND NEXT BLANK OR COMMA DELIMITER                         PARSE     974
        20 DO 30 K=J,80                                                 PARSE     975
           IF (ARRAY(K) .EQ. IBLK .OR. ARRAY(K) .EQ. ICOMMA             PARSE     976
          X .OR. ARRAY(K) .EQ. IEQ) GO TO 40                            PARSE     977
        30 CONTINUE                                                     PARSE     978
     C                                                                  PARSE     979
     C       BLANK DELIMITED CHARACTERS OCCUPY POSITIONS J TO K-1       PARSE     980
        40 IFROM = K+1                                                  PARSE     981
           L = K-1                                                      PARSE     982
           K = K-J                                                      PARSE     983
           IWORD = IBLK                                                 PARSE     984
     C                                                                  PARSE     985
     C       USE ONLY THE TEN LEFTMOST CHARACTERS WHEN BUILDING WORD    PARSE     986
           IF (K .GT. 10) L = J+9                                       PARSE     987
           ENCODE (10,55,IWORD)(ARRAY(K),K=J,L)                         PARSE     988
        55 FORMAT(10A1)                                                 PARSE     989
           RETURN                                                       PARSE     990
           END                                                          PARSE     991
```

```
      SUBROUTINE SCANE(IFROM,ARRAY,VAL,ITYPE)                                  PARSE     992
     C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C- PARSE     993
     C     THIS SUBROUTINE SCANS AN INPUT ARRAY FOR A FIELD OF CONTINUOUS       PARSE     994
     C     CHARACTERS AND EVALUATES THE LEFTMOST 20 UNDER A (E20.0) FORMAT      PARSE     995
     C                                                                          PARSE     996
     C     IFROM   -  THE POINT WHERE SCANNING BEGINS, PASSED IN                PARSE     997
     C             -  ONE PAST THE LAST POSITION SCANNED, PASSED OUT            PARSE     998
     C     ARRAY  - ARRAY IN WHICH INPUT RECORD HAS BEEN READ                   PARSE     999
     C     VAL    - WORD CONTAINING VALUE OF SCANNED CHARACTERS                 PARSE    1000
     C     ITYPE  - RETURNED  1  IMPLIES  END OF SCAN                           PARSE    1001
     C                        2           VALUE CONVERTED                      PARSE    1002
     C                        3           ERROR IN EVALUATING                  PARSE    1003
     C                        4           ALL FOUND                            PARSE    1004
     C                        5           TOL FOUND                            PARSE    1005
     C                        6           OFF FOUND                            PARSE    1006
     C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C- PARSE    1007
           INTEGER ARRAY(1),IWORD(2),ZTRUNC                                     PARSE    1008
     C                                                                          PARSE    1009
     C        ERROR COMMON WITH ERROR SUBROUTINE                                PARSE    1010
           COMMON/ERROR/CHECK                                                   PARSE    1011
           LOGICAL CHECK                                                        PARSE    1012
           DATA IBLK/10H          /,ICOMMA/10H,                                 PARSE    1013
           DATA IEQ/10H=          /                                             PARSE    1014
           IF (IFROM .GT. 80) GO TO 15                                          PARSE    1015
           ITYPE = 2                                                            PARSE    1016
     C                                                                          PARSE    1017
     C        FIND FIRST NON-BLANK, NON-COMMA CHARACTER                         PARSE    1018
           DO 10 J=IFROM,80                                                     PARSE    1019
           IF (ARRAY(J) .NE. IBLK .AND. ARRAY(J) .NE. ICOMMA                    PARSE    1020
          X .AND. ARRAY(J) .NE. IEQ) GO TO 20                                   PARSE    1021
        10 CONTINUE                                                             PARSE    1022
     C                                                                          PARSE    1023
     C        IF ITYPE EQUALS ONE THEN NO CHARACTERS REMAIN                     PARSE    1024
        15 ITYPE = 1                                                            PARSE    1025
           RETURN                                                               PARSE    1026
     C                                                                          PARSE    1027
     C        FIND NEXT BLANK OR COMMA DELIMITER                                PARSE    1028
        20 DO 30 K=J,80                                                         PARSE    1029
           IF (ARRAY(K) .EQ. IBLK .OR. ARRAY(K) .EQ. ICOMMA                     PARSE    1030
          X .OR. ARRAY(K) .EQ. IEQ) GO TO 40                                    PARSE    1031
        30 CONTINUE                                                             PARSE    1032
     C                                                                          PARSE    1033
     C        BLANK DELIMITED CHARACTERS OCCUPY POSITIONS J TO K-1              PARSE    1034
        40 IFROM = K+1                                                          PARSE    1035
           L = K-1                                                              PARSE    1036
           K = K-J                                                              PARSE    1037
           IWORD(1) = IWORD(2) = IBLK                                           PARSE    1038
     C                                                                          PARSE    1039
     C        USE ONLY THE TWENTY LEFTMOST CHARACTERS WHEN BUILDING WORD        PARSE    1040
           IF (K .GT. 20) L = J+19                                              PARSE    1041
           ENCODE (10,55,IWORD)(ARRAY(K),K=J,L)                                 PARSE    1042
        55 FORMAT(10A1)                                                         PARSE    1043
           IARG=ZTRUNC(IWORD(1),1)                                              PARSE    1044
           IF(IARG .NE. 1LT) GO TO 60                                           PARSE    1045
           ITYPE = 5                                                            PARSE    1046
           RETURN                                                               PARSE    1047
        60 IF(IARG .NE. 1LA) GO TO 80                                           PARSE    1048
```

```
                ITYPE = 4                                                       PARSE   1049
                RETURN                                                          PARSE   1050
60          80  IF(IARG .NE. 1LO) GO TO 90                                     PARSE   1051
                ITYPE = 6                                                       PARSE   1052
                RETURN                                                          PARSE   1053
            C                                                                   PARSE   1054
            C       EVALUATE CHARACTER STREAM                                   PARSE   1055
65          90  DECODE (20,95,IWORD) VAL                                       PARSE   1056
            95  FORMAT(E20.0)                                                  PARSE   1057
                IF (    .NOT.CHECK) GO TO 99999                                 PARSE   1058
                ITYPE = 3                                                       PARSE   1059
                CALL ERROR4(25H(20H INVALID NUMERIC ARG),2,5HSCANE)             PARSE   1060
70              CHECK = .FALSE.                                                 PARSE   1061
            99999 RETURN                                                        PARSE   1062
                END                                                             PARSE   1063
```

```
      SUBROUTINE SQUEEZE(IREC)                                          PARSE    1064
C                                                                       PARSE    1065
C       THIS SUBROUTINE REMOVES UNNECESSARY BLANKS FROM THE ARRAY PASSE PARSE    1066
C       IN. ARRAY MUST HAVE BEEN READ UNDER AN (80A1) FORMAT.           PARSE    1067
C                                                                       PARSE    1068
      DIMENSION IREC(1)                                                 PARSE    1069
      LOGICAL IBSWT                                                     PARSE    1070
      I = 1                                                             PARSE    1071
      LPS = 1                                                           PARSE    1072
   10 IBSWT = .FALSE.                                                   PARSE    1073
C                                                                       PARSE    1074
C       SKIP KEYWORD                                                    PARSE    1075
      CALL SCAN(LPS,IREC,IWORD)                                         PARSE    1076
      DO 30 J=LPS,80                                                    PARSE    1077
   15 IF (IREC(J) .EQ. 1H ) GO TO 20                                    PARSE    1078
      IREC(I) = IREC(J)                                                 PARSE    1079
      I = I + 1                                                         PARSE    1080
      IBSWT = .FALSE.                                                   PARSE    1081
      GO TO 30                                                          PARSE    1082
   20 IF (IBSWT) GO TO 30                                               PARSE    1083
      IBSWT = .TRUE.                                                    PARSE    1084
      IREC(I) = IREC(J)                                                 PARSE    1085
      I = I + 1                                                         PARSE    1086
   30 CONTINUE                                                          PARSE    1087
   25 C                                                                 PARSE    1088
C       BLANK OUT REMAINDER OF INPUT RECORD                             PARSE    1089
      IF (I .GT. 80) GO TO 70                                           PARSE    1090
      DO 60 J=I,80                                                      PARSE    1091
   60 IREC(J) = 1H                                                      PARSE    1092
   30 70 RETURN                                                         PARSE    1093
      END                                                               PARSE    1094
```

```
      SUBROUTINE TITLE                                                  PARSE   1095
C                                                                       PARSE   1096
C        THIS SUBROUTINE CONTROLS THE PROCESSING OF TEXT ON A TITLE CARD PARSE   1097
C        A TEXT BEAD IS CREATED, FILLED, AND ATTACHED TO THE TITLE TEXT  PARSE   1098
C        LIST EMMITTING FROM THE SECOND WORD OF THE CURRENT PLOT GROUP   PARSE   1099
C        BEAD, PROVIDED THE LIMIT OF TITLE CARDS IS NOT EXCEEDED.       PARSE   1100
C                                                                       PARSE   1101
      COMMON/SVECTOR/LPG,LAXIS,LPM,IESWT,LCONCRD                        PARSE   1102
      COMMON/CCODES/IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6     PARSE   1103
      DATA LIMARB/3/                                                    PARSE   1104
C                                                                       PARSE   1105
C        CREATE AND FILL TEXT BEAD                                      PARSE   1106
      CALL REFORM(IPTR,2)                                               PARSE   1107
C                                                                       PARSE   1108
C        ATTACH TEXT BEAD FROM CURRENT P-G BEAD                         PARSE   1109
      CALL DMGET(IWHOLE2,LPG,INPTR)                                     PARSE   1110
      NOFCDS = 0                                                        PARSE   1111
C                                                                       PARSE   1112
C        DETERMINE IF LIST IS NULL                                      PARSE   1113
      IF (INPTR .NE. 0) GO TO 10                                        PARSE   1114
C                                                                       PARSE   1115
C        ATTACH DIRECTLY FROM P-G BEAD                                  PARSE   1116
      CALL DMSET(IWHOLE2,LPG,IPTR)                                      PARSE   1117
      RETURN                                                            PARSE   1118
C                                                                       PARSE   1119
C        ATTACH FROM LAST TITLE BEAD                                    PARSE   1120
   10 NPTR = INPTR                                                      PARSE   1121
      NOFCDS = NOFCDS + 1                                               PARSE   1122
      CALL DMGET(IWHOLE1,NPTR,INPTR)                                    PARSE   1123
      IF (INPTR .NE. 0) GO TO 10                                        PARSE   1124
      IF (NOFCDS .GE. LIMARB) GO TO 9991                                PARSE   1125
      CALL DMSET(IWHOLE1,NPTR,IPTR)                                     PARSE   1126
      RETURN                                                            PARSE   1127
C                                                                       PARSE   1128
C        ERROR CALLS                                                    PARSE   1129
 9991 CALL ERRORM(31H(26H TITLE CARD LIMIT EXCEEDED), 1,5HTITLE)        PARSE   1130
      RETURN                                                            PARSE   1131
      END                                                               PARSE   1132
```

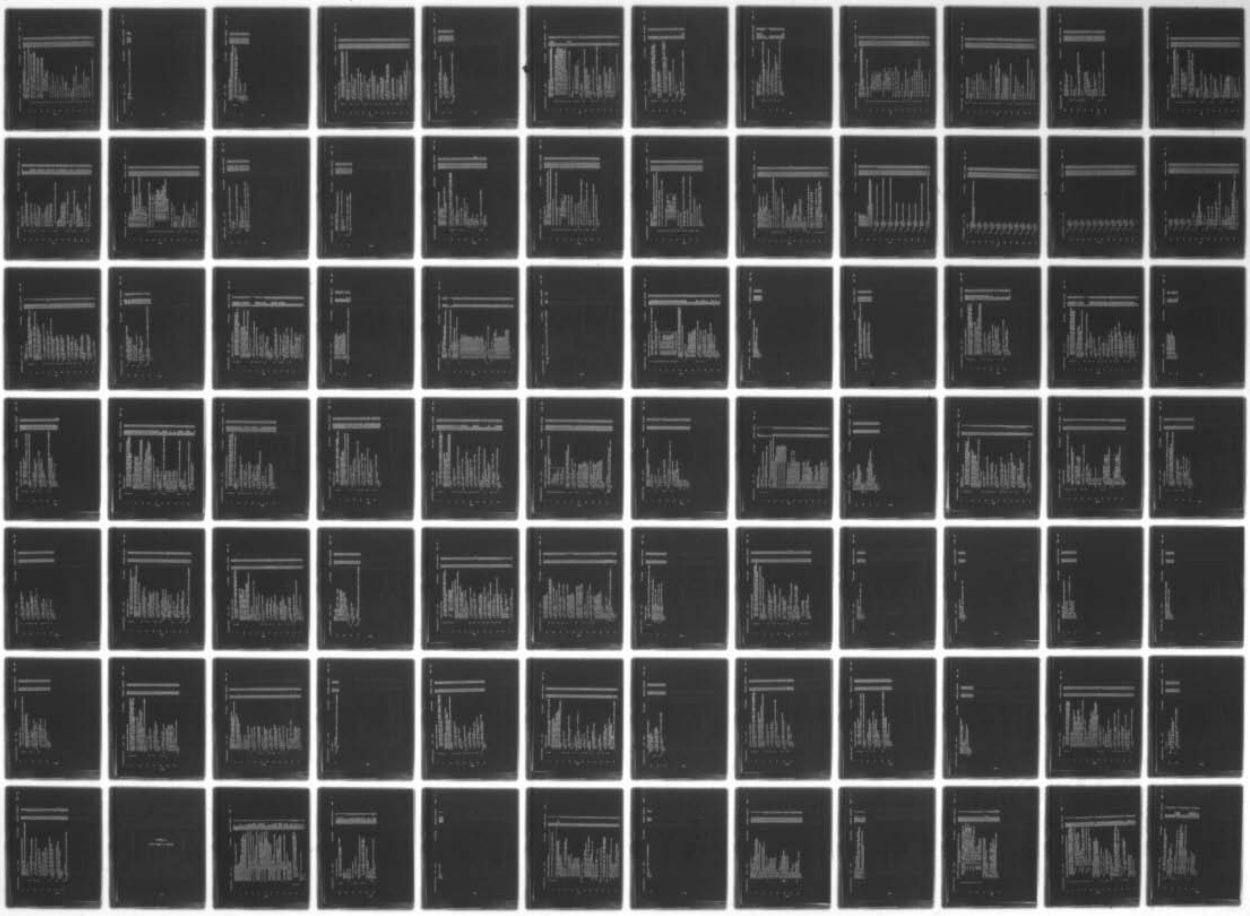
```
      1                 OVERLAY(4,0)                                                  COLLECT     2
                        SUBROUTINE FN(IFNPT,TEST,VAL)                                 COLLECT     3
                        LOGICAL TEST                                                  COLLECT     4
                        INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT   5
      5                 COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT 6
                        COMMON /STATUS/ STARR(10)                                     COLLECT     7
                  C                                                                   COLLECT     8
                  C     SUBROUTINE EVALUATES FUNCTIONS - ELEMENT VALUE                COLLECT     9
                  C     VERSUS PREDEFINED ARGUMENTS                                   COLLECT    10
     10           C                                                                   COLLECT    11
                  C     IFNPT IS A POINTER TO THE BEAD WHICH CONTAINS                 COLLECT    12
                  C     A DESCRIPTION OF THE FUNCTION TO BE APPLIED, ITS              COLLECT    13
                  C     PREDEFINED ARGUMENTS, ETC.                                    COLLECT    14
                  C     TEST IS SET TO FALSE IF THE CRITERIA ARE                      COLLECT    15
     15           C     UNSATISFIED, TRUE IF THEY ARE SATISFIED                       COLLECT    16
                  C     VAL ENTERS AS THE VALUE OF THE ELEMENT TO                     COLLECT    17
                  C     BE TESTED. VAL RETURNS AS THE ADJUSTED VALUE                  COLLECT    18
                  C     WHICH SATISFIED THE FUNCTION                                  COLLECT    19
                  C                                                                   COLLECT    20
     20           C     FUNCTION BEAD FORMAT                                          COLLECT    21
                  C           * NEXT FUNCTION BEAD PTR *                            COLLECT    22
                  C           * FUNCTION NAME          *                            COLLECT    23
                  C           * NO OF ARGS             *                            COLLECT    24
                  C           * TOLERANCE              *                            COLLECT    25
     25           C           * ARG1                   *                            COLLECT    26
                  C           *                        *                            COLLECT    27
                  C           *   ETC                  *                            COLLECT
                  C           *                        *                            COLLECT
                  C           ************************                              COLLECT    30
     30           C                                                                   COLLECT    31
                  C     FUNCTIONS ARE LE,LT,GE,GT,EQ,NE,RANGE                         COLLECT    32
                  C     CODES ARE      1, 2, 3, 4, 5, 6,  7                           COLLECT    33
                  C                                                                   COLLECT    34
                  C     DECODE FUNCTION TO BE APPLIED                                 COLLECT    35
     35           C                                                                   COLLECT    36
                        CALL DMGET(WHOL2,IFNPT,IFNCOD)                                COLLECT    37
                  C                                                                   COLLECT    38
                  C     GET NO OF ARGUMENTS                                           COLLECT    39
                  C                                                                   COLLECT    40
     40                 CALL DMGET(WHOL3,IFNPT,NOOFAG)                               COLLECT    41
                  C                                                                   COLLECT    42
                  C     GET TOLERANCE                                                 COLLECT    43
                  C                                                                   COLLECT    44
                        CALL DMGET(WHOL4,IFNPT,TOL)                                   COLLECT    45
     45           C                                                                   COLLECT    46
                  C     CHECK FOR SPECIAL ALL CASE                                    COLLECT    47
                  C                                                                   COLLECT    48
                        IF(NOOFAG .GT. 0) GO TO 5                                     COLLECT    49
                        IF(IFNCOD .EQ. 5) GO TO 155                                   COLLECT    50
     50                 IF(IFNCOD .EQ. 6) GO TO 160                                  COLLECT    51
                        GO TO 170                                                     COLLECT    52
                  C                                                                   COLLECT    53
                  C     GET ITH ARGUMENT                                              COLLECT    54
                  C                                                                   COLLECT    55
     55             5 CONTINUE                                                        COLLECT    56
                       I=1                                                            COLLECT    57
                    10 CONTINUE                                                       COLLECT    58
```

```
                      CALL OMGET(HHOL4+ICT,IFNPT,ARGI)                          COLLECT    59
                C                                                               COLLECT    60
     60         C     SELECT FUNCTION                                           COLLECT    61
                C                                                               COLLECT    62
                      IF((IFNCOD .LT. 1).OR.(IFNCOD .GT. 10)) GO TO 170         COLLECT    63
                      GO TO (20,30,40,50,60,70,80,90,100,110),IFNCOD            COLLECT    64
                C                                                               COLLECT    65
     65         C     FUNCTION LE                                               COLLECT    66
                C                                                               COLLECT    67
                 20   CONTINUE                                                  COLLECT    68
                      IF(VAL .LE. ARGI) GO TO 155                               COLLECT    69
                      GO TO 160                                                 COLLECT    70
     70         C                                                               COLLECT    71
                C     FUNCTION LT                                               COLLECT    72
                C                                                               COLLECT    73
                 30   CONTINUE                                                  COLLECT    74
                      IF(VAL .LT. ARGI) GO TO 155                               COLLECT    75
     75               GO TO 160                                                 COLLECT    76
                C                                                               COLLECT    77
                C     FUNCTION GE                                               COLLECT    78
                C                                                               COLLECT    79
                 40   CONTINUE                                                  COLLECT    80
     80               IF(VAL .GE. ARGI) GO TO 155                               COLLECT    81
                      GO TO 160                                                 COLLECT    82
98              C                                                               COLLECT    83
                C     FUNCTION GT                                               COLLECT    84
                C                                                               COLLECT    85
     85          50   CONTINUE                                                  COLLECT    86
                      IF(VAL .GT. ARGI) GO TO 155                               COLLECT    87
                      GO TO 160                                                 COLLECT    88
                C                                                               COLLECT    89
                C     FUNCTION EQ                                               COLLECT    90
     90         C                                                               COLLECT    91
                 60   CONTINUE                                                  COLLECT    92
                      IF((VAL .LE. ARGI+TOL) .AND. (VAL .GE. ARGI-TOL))         COLLECT    93
                     1GO TO 150                                                 COLLECT    94
                      ICT=ICT+1                                                 COLLECT    95
     95               IF(ICT .GT. NOOFAG)GO TO 160                              COLLECT    96
                      GO TO 10                                                  COLLECT    97
                C                                                               COLLECT    98
                C     FUNCTION NE                                               COLLECT    99
                C                                                               COLLECT   100
    100          70   CONTINUE                                                  COLLECT   101
                      IF(VAL .EQ. ARGI) GO TO 160                               COLLECT   102
                      ICT=ICT+1                                                 COLLECT   103
                      IF(ICT .GT. NOOFAG) GO TO 155                             COLLECT   104
                      GO TO 10                                                  COLLECT   105
    105         C                                                               COLLECT   106
                C     FUNCTION RANGE                                            COLLECT   107
                C                                                               COLLECT   108
                 80   CONTINUE                                                  COLLECT   109
                      GO TO (82,83),ICT                                         COLLECT   110
    110            82 IF(VAL .GE. ARGI) GO TO 85                                COLLECT   111
                      GO TO 160                                                 COLLECT   112
                   83 IF(VAL .LE. ARGI) GO TO 155                               COLLECT   113
                      GO TO 160                                                 COLLECT   114
                 85   ICT=2                                                     COLLECT   115
```

```
      GO TO 10                                                          COLLECT  116
C                                                                       COLLECT  117
C     SPACE FOR EXTRA FUNCTION                                          COLLECT  118
C                                                                       COLLECT  119
 90   CONTINUE                                                          COLLECT  120
      GO TO 170                                                         COLLECT  121
C                                                                       COLLECT  122
C     SPACE FOR EXTRA FUNCTION                                          COLLECT  123
C                                                                       COLLECT  124
 100  CONTINUE                                                          COLLECT  125
      GO TO 170                                                         COLLECT  126
C     SPACE FOR EXTRA FUNCTION                                          COLLECT  127
C                                                                       COLLECT  128
 110  CONTINUE                                                          COLLECT  129
      GO TO 170                                                         COLLECT  130
C                                                                       COLLECT  131
C     DONE, ADJUST FINAL VALUE TO MATCH TOLERANCE                       COLLECT  132
C     AND RETURN, SUCCESS                                               COLLECT  133
C                                                                       COLLECT  134
 150  CONTINUE                                                          COLLECT  135
      VAL=ARG1                                                          COLLECT  136
 155  TEST=.TRUE.                                                       COLLECT  137
      RETURN                                                            COLLECT  138
C                                                                       COLLECT  139
C     FAIL                                                              COLLECT  140
C                                                                       COLLECT  141
 160  CONTINUE                                                          COLLECT  142
      TEST=.FALSE.                                                      COLLECT  143
      RETURN                                                            COLLECT  144
C                                                                       COLLECT  145
C     INVALID FUNCTION **ERROR**                                        COLLECT  146
C                                                                       COLLECT  147
 170  CONTINUE                                                          COLLECT  148
      CALL ERRORM(26H(21H INV FUN, CALL ANLYST),-3,2HFN)                COLLECT  149
      END                                                               COLLECT  150
```

AD-A058 673 DAYTON UNIV OHIO RESEARCH INST F/G 9/2
PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). VOLUME II--ETC(U)
MAY 78 J G JENSEN F33615-76-C-3076
UNCLASSIFIED UDR-TR-78-34 AFFDL-TR-78-55-VOL-2 NL

2 OF 7
AD A058 673



```
      1                SUBROUTINE FNCTEV(IFNPTR,TEST,VALNAM)                                  COLLECT   151
                       INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50        COLLECT   152
                       LOGICAL TEST                                                           COLLECT   153
                       COMMON /STATUS/ STARR(10)                                              COLLECT   154
      5                COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50     COLLECT   155
                  C                                                                           COLLECT   156
                  C    SUBROUTINE EVALUATES ALL FUNCTIONS ASSOCIATED WITH CURRENT             COLLECT   157
                  C    ELEMENT NAME GIVEN BY IFNPTR                                           COLLECT   158
                  C    SUBROUTINE IS CALLED BY SCANCH TO EVALUATE CONDITION ELEMENTS          COLLECT   159
     10           C    AND BY PH GRUP TO EVALUATE PARAMETER AND MULTIPLIER ELEMENTS           COLLECT   160
                  C                                                                           COLLECT   161
                  C    IFNPTR IS POINTER TO READ WHICH CONTAINS ELEMENT NAME                  COLLECT   162
                  C    RESULT IS LOGICAL RESULT OF ALL TESTS ON THE CURRENT                   COLLECT   163
                  C    ELEMENT NAME                                                           COLLECT   164
     15           C    VALNAME IS VALUE OF ELEMENT NAME WHICH SATISFIED TEST                  COLLECT   165
                  C    ELEMENT READ (CONDITION, PARAMETER, MULTIPLIER) FORMAT                 COLLECT   166
                  C            * NEXT ELEMENT PTR          *                                 COLLECT   167
                  C            * CODE FOR ELEMENT NAME     *                                 COLLECT   168
                  C            * PTR TO LIST OF FUNCTIONS  *                                 COLLECT   169
     20           C            ******************************                                COLLECT   170
                  C                                                                           COLLECT   171
                  C    GET 1ST FUNCTION                                                       COLLECT   172
                  C                                                                           COLLECT   173
                       CALL DMGET(WHOL3,IFNPTR,STARR(4))                                      COLLECT   174
88    25         C                                                                           COLLECT   175
                  C    IF NO FUNCTION, ERROR                                                  COLLECT   176
                  C                                                                           COLLECT   1
                       IF(STARR(4) .EQ. 0) GO TO 40                                           COLLECT   1
                  C                                                                           COLLECT   179
     30           C    GET ELEMENT NAME                                                       COLLECT   180
                  C                                                                           COLLECT   181
                       CALL DMGET(WHOL2,IFNPTR,NAME)                                          COLLECT   182
                  C                                                                           COLLECT   183
                  C       GET ELEMENT VALUE                                                   COLLECT   184
     35                VALNAM = GETVAL(NAME)                                                  COLLECT   185
                  C                                                                           COLLECT   186
                  C    FUNCTION LOOP                                                          COLLECT   187
                  C                                                                           COLLECT   188
                    20 CONTINUE                                                               COLLECT   189
     40                TEST=.FALSE.                                                           COLLECT   190
                    30 CALL FN(STARR(4),TEST,VALNAM)                                          COLLECT   191
                  C                                                                           COLLECT   192
                  C    IF TEST SATISFIED, RETURN FOR NEXT ELEMENT                             COLLECT   193
                  C                                                                           COLLECT   194
     45           C    IF(TEST) RETURN                                                        COLLECT   195
                  C                                                                           COLLECT   196
                  C    GET NEXT FUNCTION                                                      COLLECT   197
                  C                                                                           COLLECT   198
                       CALL DMGET(WHOL1,STARR(4),STARR(4))                                    COLLECT   199
     50           C                                                                           COLLECT   200
                  C    CHECK FOR NO MORE FUNCTIONS                                            COLLECT   201
                  C                                                                           COLLECT   202
                       IF(STARR(4) .EQ. 0) RETURN                                             COLLECT   203
                       GO TO 30                                                               COLLECT   204
     55           C                                                                           COLLECT   205
                  C    NO FUNCTIONS TO EVALUATE FOR THIS ELEMENT **ERROR**                    COLLECT   206
                  C                                                                           COLLECT   207
```

```
      40    CONTINUE                                                          COLLECT   208
            CALL ERRORM(32H(27H NO FUN TO EVL, CALL ANLYST),-3,6HFNCTEV)       COLLECT   209
      60    END                                                               COLLECT   210
```

```
      REAL FUNCTION GETVAL(NAME)                                        COLLECT  211
      INTEGER WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50         COLLECT  212
      COMMON/COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT  213
C                                                                       COLLECT  214
C        THIS ROUTINE OBTAINS CORRECT ELEMENT VALUES FROM THE CURRENT   COLLECT  215
C        INPUT RECORD, GIVEN THE ELEMENT CODE NUMBER (NAME).            COLLECT  216
C                                                                       COLLECT  217
      IF(NAME .GT. 500) GO TO 30                                        COLLECT  218
      GETVAL = NODVAL(NAME)                                             COLLECT  219
      RETURN                                                            COLLECT  220
   30 CONTINUE                                                          COLLECT  221
      GETVAL = SPCVAL(NAME)                                             COLLECT  222
      RETURN                                                            COLLECT  223
      END                                                               COLLECT  224
```

```
      SUBROUTINE L8LSET(IPLOT)                                          COLLECT  225
C                                                                       COLLECT  226
C        THIS ROUTINE INITIALIZES OR UPDATES THE MAX AND MIN VALUES OF  COLLECT  227
C        REQUESTED LABEL ELEMENTS, POINTED TO BY THE 10TH WORD OF THE   COLLECT  228
C        CURRENT PLOT BEAD (IPLOT).                                     COLLECT  229
C                                                                       COLLECT  230
      COMMON/COMPCOD/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,  COLLECT  231
     XIARRY20,IARRY50                                                   COLLECT  232
      COMMON/STATUS/ STARR(10)                                          COLLECT  233
C                                                                       COLLECT  234
C        GET POINTER TO LABEL MXMN BEADS                                COLLECT  235
      ICOMP = IWHOLE0 +10                                               COLLECT  236
      CALL DMGET(ICOMP,IPLOT,IMXMN)                                     COLLECT  237
      IF (IMXMN .NE. 0) GO TO 20                                        COLLECT  238
C                                                                       COLLECT  239
C        NO L8L BDS, INITIALIZE IF REQUEST MADE                         COLLECT  240
      LAST = 0                                                          COLLECT  241
      CALL DMGET(IWHOLE0+7,STARR(2),IREQ)                               COLLECT  242
   10 IF (IREQ .EQ. 0) RETURN                                           COLLECT  243
C                                                                       COLLECT  244
C        DO, FOR EACH L8L REQUEST BEAD                                  COLLECT  245
C                                                                       COLLECT  246
C        CREATE LABEL MAX&MIN                                           COLLECT  247
      CALL DMGTBD(4,IMXMN)                                              COLLECT  248
      IF (IMXMN .EQ. 0) GO TO 9991                                      COLLECT  249
C                                                                       COLLECT  250
C        GET ELE NO. & STORE INTO MX-MN BEAD                            COLLECT  251
      CALL DMGET(IWHOLE2,IREQ,IELENO)                                   COLLECT  252
      CALL DMSET(IWHOLE2,IMXMN,IELENO)                                  COLLECT  253
C                                                                       COLLECT  254
C        GET VALUE OF ELE NO.                                           COLLECT  255
      VAL = GETVAL(IELENO)                                              COLLECT  256
C                                                                       COLLECT  257
C        STORE VALUE IN MAX-MIN BEAD                                    COLLECT  258
      CALL DMSET(IWHOLE3,IMXMN,VAL)                                     COLLECT  259
      CALL DMSET(IWHOLE4,IMXMN,VAL)                                     COLLECT  260
C                                                                       COLLECT  261
C        STRING ON MAX-MIN BEAD                                         COLLECT  262
      IF (LAST .EQ. 0) CALL DMSET(ICOMP,IPLOT,IMXMN)                    COLLECT  263
      IF (LAST .NE. 0) CALL DMSET(IWHOLE1,LAST,IMXMN)                   COLLECT  264
      LAST = IMXMN                                                      COLLECT  265
C                                                                       COLLECT  266
C        GET NEXT L8L REQUEST BEAD                                      COLLECT  267
      CALL DMGET(IWHOLE1,IREQ,IREQ)                                     COLLECT  268
      GO TO 10                                                          COLLECT  269
C                                                                       COLLECT  270
C        UPDATE LABEL MX-MN BEADS                                       COLLECT  271
C                                                                       COLLECT  272
C        GET ELEMENT NO. & VALUE                                        COLLECT  273
   20 CALL DMGET(IWHOLE2,IMXMN,IELENO)                                  COLLECT  274
      VAL = GETVAL(IELENO)                                              COLLECT  275
C                                                                       COLLECT  276
C        GET MAX VALUE & UPDATE IF NECESSARY                            COLLECT  277
      CALL DMGET(IWHOLE3,IMXMN,ELEVAL)                                  COLLECT  278
      IF  (ELEVAL .LT. VAL) CALL DMSET(IWHOLE3,IMXMN,VAL)               COLLECT  279
C                                                                       COLLECT  280
C        GET MIN VAL AND UPDATE IF NECESSARY                            COLLECT  291
```

```
      CALL DMGET(IWHOLE4,IMXMN,ELEVAL)                                  COLLECT  282
      IF (ELEVAL .GT. VAL) CALL DMSET(IWHOLE4,IMXMN,VAL)                COLLECT  283
                                                                        COLLECT  284
60    C                                                                 COLLECT  285
      C     GET NEXT LBL MX-MN BEAD                                     COLLECT  286
      CALL DMGET(IWHOLE1,IMXMN,IMXMN)                                   COLLECT  287
      IF (IMXMN .EQ. 0) RETURN                                          COLLECT  288
      GO TO 20                                                          COLLECT  289
65    C                                                                 COLLECT  290
      C     ERROR CALLS                                                 COLLECT  291
 9991 CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,6HLBLSET)            COLLECT  292
      END
```

```
      1                SUBROUTINE MULSRH(IPLOT)                                  MODA      126
                       INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50  COLLECT   294
                       COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT 295
                       COMMON /STATUS/ STARR(10)                                 COLLECT   296
      5          C                                                               COLLECT   297
                 C        THIS SUBROUTINE SEARCHES THE PLOT BEAD LIST FOR BEADS   MODA      127
                 C        CONTAINING VALUES WHICH MATCH THE CURRENT DATA RECORD AT ALL  MODA 128
                 C        MULTIPLIER DEPTHS. IPLOT IS SUPPLIED AS THE BEGINNING POINTER TO MODA 129
                 C        THE PLOT BEAD LIST, AND RETURNED AS A POINTER TO A NEW OR OLD  MODA 130
     10          C        PLOT BEAD WHICH SATISIFIES BOTH THE MULTIPLIER SPECIFICATIONS  MODA 131
                 C        OF THE CURRENT P-M GROUP AND THE VALUES OF THE CURRENT INPUT  MODA 132
                 C        RECORD. IF A NEW PLOT BEAD IS ALLOCATED, THEN IT IS INSERTED  MODA 133
                 C        IN SORTED ORDER. THE PRIMARY SORT KEY IS THE VALUE OF THE  MODA 134
                 C        FIRST LEVEL MULTIPLIER. THE SECONDARY SORT KEY IS THE VALUE OF  MODA 135
     15          C        THE SECOND LEVEL MULTIPLIER, ETC.                       MODA      136
                 C                                                                COLLECT   312
                 C     THE DEPTH COUNT IS IN STARR(8), THE MUL VALUE PTR IN STARR(9)  COLLECT 313
                 C                                                                COLLECT   314
                       IOPLOT = 0                                                 MODA      137
     20                IF (IPLOT .EQ. 0) GO TO 30                                 MODA      138
                 C                                                                MODA      139
                 C         LOOP THROUGH THE PLOT BEADS AND SEARCH                 MODA      140
                 C         FOR A MULTIPLIER MATCH AT ALL LEVELS.                  MODA      141
                 C         THE LOOP ENDS WITH THE STATEMENT "GO TO 10".           MODA      142
     25          C                                                                MODA      143
                       LENGTH = STARR(8)                                          MODA      144
                    10 DO 15 ILEV=1,LENGTH                                        MODA      145
                 C                                                                MODA      146
                 C        GET RECORD MULT VALUE AND COMPARE WITH EXISTING PLOT BD MULT  MODA 147
                       CALL DMGET(WHOL0+ILEV,STARR(9),VALIN)                      MODA      148
     30                CALL DMGET(WHOL0+12+2*ILEV,IPLOT,TVAL)                     MODA      149
                       IF (VALIN .GT. TVAL) GO TO 20                              MODA      150
                       IF (VALIN .LT. TVAL) GO TO 30                              MODA      151
                    15 CONTINUE                                                   MODA      152
     35          C                                                                MODA      153
                 C        MATCH FOUND, RETURN PLOT BD POINTER                    MODA      154
                       RETURN                                                     MODA      155
                 C                                                                MODA      156
                 C        MATCH NOT FOUND, UPDATE PLOT BD POINTER AND CONTINUE IF NONZERO  MODA 157
     40             20 IOPLOT = IPLOT                                             MODA      158
                       CALL DMGET(WHOL1,IOPLOT,IPLOT)                             MODA      159
                       IF (IPLOT .EQ. 0) GO TO 30                                 MODA      160
                       GO TO 10                                                   MODA      161
                 C                                                                MODA      162
     45          C        ADD A NEW PLOT BEAD TO THE STRUCTURE                    MODA      163
                    30 CALL DMGET(12+2*STARR(8),NBEAD)                            MODA      164
                       IF (NBEAD .EQ. 0) GO TO 9992                               MODA      165
                       DO 35 ILEV=1,LENGTH                                        MODA      166
                       CALL DMGET(WHOL0+ILEV,STARR(9),VALIN)                      MODA      167
     50                CALL DMSET(WHOL0+12+2*ILEV,NBEAD,VALIN)                    MODA      168
                    35 CONTINUE                                                   MODA      169
                       CALL DMSET(WHOL0+12,NBEAD,LENGTH)                          MODA      170
                       IF (IOPLOT .NE. 0) GO TO 40                                MODA      171
                 C                                                                MODA      172
     55          C         SET INTO THE P-M BEAD A POINTER TO THE NEW PLOT BEAD.   MODA      173
                 C                                                                MODA      174
                       CALL DMSET(WHOLE,STARR(6),NBEAD)                           MODA      175
```

```
                      GO TO 50                                                  MODA      176
                C                                                               MODA      177
60              C         SET INTO THE PREVIOUS PLOT BEAD A FORWARD, "CROSS PLOT"  MODA     178
                C         POINTER TO THE NEW PLOT BEAD. SET INTO THE NEW PLOT BEAD A MODA     179
                C         BACKWARD, "CROSS PLOT" POINTER TO THE PREVIOUS PLOT BEAD. MODA      180
                C                                                               MODA      181
                   40 CALL DMSET(WHOL1,IOPLOT,NBEAD)                            MODA      182
65                    CALL DMSET(WHOL3,NBEAD,IOPLOT)                            MODA      183
                   50 IF(IPLOT .EQ. 0) GO TO 60                                 MODA      184
                C                                                               MODA      185
                C         SET INTO THE NEW PLOT BEAD A FORWARD, "CROSS PLOT" POINTER TO MODA 186
                C         THE NEXT PLOT BEAD. SET INTO THE NEXT PLOT BEAD A BACKWARD, MODA 187
70              C         "CROSS PLOT" POINTER TO THE NEW PLOT BEAD.            MODA      188
                C                                                               MODA      189
                      CALL DMSET(WHOL1,NBEAD,IPLOT)                             MODA      190
                      CALL DMSET(WHOL3,IPLOT,NBEAD)                             MODA      191
                C                                                               MODA      192
75              C         RETURN THE POINTER TO THE NEW PLOT BEAD.              MODA      193
                C                                                               MODA      194
                   60 IPLOT = NBEAD                                             MODA      195
                      RETURN                                                    MODA      196
                C                                                               MODA      197
80              C         ERROR MESSAGES                                        MODA      198
                 9992 CONTINUE                                                  MODA      199
                      CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,6HMULSRH)    COLLECT   384
                      END                                                       COLLECT   385
```

```
      SUBROUTINE PLOT8D                                                 COLLECT   386
      INTEGER STARR                                                     COLLECT   387
      COMMON /STATUS/ STARR(10)                                         COLLECT   388
C                                                                       COLLECT   389
C     THIS ROUTINE IS CALLED TO SEARCH AND PROCESS THE STRUCTURE BELOL  MODA      200
C     THE CURRENT PARAMETER-MULTIPLIER BEAD.                            MODA      201
C                                                                       PM          2
C        PARAMETER - MULTIPLIER BEAD                                    PM          3
      COMMON /PM/ IPMNXT,IPMLST,IPMPL,IPMML,IPMPB,IPMPT,IPMMT           PM          4
C                                                                       MODA      203
C        GET THE FIRST PLOT BEAD                                        MODA      204
      CALL DMGET(IPMPB,STARR(6),IPLOT)                                  MODA      205
C                                                                       MODA      206
C        FIND MATCHING PLOT BEAD, IF MULTIPLIERS EXIST                  MODA      207
      IF (STARR(8) .NE. 0) CALL MULSRH(IPLOT)                           MODA      208
C                                                                       MODA      209
C        SEARCH THE CURVE BEAD LIST FOR THE LOCATIONS TO STORE X-Y VALS  MODA      210
      CALL FRMSRH(IPLOT)                                                MODA      211
      RETURN                                                            COLLECT   407
      END                                                               COLLECT   408
```

```
      SUBROUTINE PM GRUP                                                    COLLECT   409
      INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50       COLLECT   410
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50    COLLECT   411
      COMMON /STATUS/ STARR(10)                                             COLLECT   412
      LOGICAL TEST                                                          COLLECT   413
C                                                                           COLLECT   414
C     SUBROUTINE DETERMINES ON WHICH PLOT TO PLACE WHICH                    COLLECT   415
C     DATA ELEMENTS                                                         COLLECT   416
C                                                                           COLLECT   417
C     AXES BEAD FORMAT                                                      COLLECT   418
C             *PTR TO NEXT AXES BEAD  *                                     COLLECT   419
C             *X AXIS ELEMENT NAME    *                                     COLLECT   420
C             * NO OF Y AXES          *                                     COLLECT   421
C             * PTR TO P-M GROUP LIST *                                     COLLECT   422
C             * Y1 AXIS ELEMENT NAME  *                                     COLLECT   423
C             * Y2 AXIS ELEMENT NAME  *                                     COLLECT   424
C             * Y3 AXIS ELEMENT NAME  *                                     COLLECT   425
C             *************************                                     COLLECT   426
C                                                                           COLLECT   427
C     P-M GROUP BEAD FORMAT                                                 COLLECT   428
C             * PTR TO NEXT P-M GROUP BEAD *                                COLLECT   429
C             * PTR TO PARAMETER LIST      *                                COLLECT   430
C             * PTR TO MULTIPLIER LIST*                                     COLLECT   431
C             * DEPTH OF MULTIPLIER LIST   *                                COLLECT   432
C             * PTR TO PLOT LIST           *                                COLLECT   433
C             ******************************                                COLLECT   434
C                                                                           COLLECT   435
C     SCAN ALL AXES AND P-M GROUP BEADS                                     COLLECT   436
C                                                                           COLLECT   437
C     INITIALIZE AXES BEAD PTR IN STATUS ARRAY                              COLLECT   438
C                                                                           COLLECT   439
      CALL DMGET(WHOL4,STARR(2),STARR(5))                                   COLLECT   440
      GO TO 15                                                              COLLECT   441
C                                                                           COLLECT   442
C     GET NEXT AXES BEAD                                                    COLLECT   443
C                                                                           COLLECT   444
   10 CONTINUE                                                              COLLECT   445
      CALL DMGET(WHOL1,STARR(5),STARR(5))                                   COLLECT   446
C                                                                           COLLECT   447
C     CHECK FOR LAST AXES BEAD                                              COLLECT   448
C                                                                           COLLECT   449
   15 CONTINUE                                                              COLLECT   450
      IF(STARR(5) .EQ. 0) RETURN                                            COLLECT   451
C                                                                           COLLECT   452
C     GET NUMBER OF Y AXES FROM AXIS BEAD                                   COLLECT   453
C                                                                           COLLECT   454
      CALL DMGET(WHOL3,STARR(5),STARR(10))                                  COLLECT   455
C                                                                           COLLECT   456
C     GET POINTER TO FIRST P-M GROUP BEAD FROM AXES BEAD                    COLLECT   457
C                                                                           COLLECT   458
      CALL DMGET(WHOL4,STARR(5),STARR(6))                                   COLLECT   459
      GO TO 25                                                              COLLECT   460
C                                                                           COLLECT   461
C     GET NEXT P-M GROUP BEAD                                               COLLECT   462
C                                                                           COLLECT   463
   20 CONTINUE                                                              COLLECT   464
      CALL DMGET(WHOL1,STARR(6),STARR(6))                                   COLLECT   465
```

```
      C                                                                 COLLECT  466
      C     CHECK FOR LAST P-M GROUP BEAD                               COLLECT  467
   60 C     IF NONE, GET NEXT AXES BEAD                                 COLLECT  468
      C                                                                 COLLECT  469
   25    CONTINUE                                                       COLLECT  470
         IF(STARR(6) .EQ. 0) GO TO 10                                   COLLECT  471
      C                                                                 COLLECT  472
   65 C     CHECK FOR PARAMETER LIST                                    COLLECT  473
      C                                                                 COLLECT  474
         CALL DMGET(WHOL2,STARR(6),STARR(7))                            COLLECT  475
         IF(STARR(7) .EQ. 0) GO TO 30                                   COLLECT  476
      C                                                                 COLLECT  477
   70 C     PARAMETER LIST IS PRESENT,EVALUATE                          COLLECT  478
      C                                                                 COLLECT  479
         CALL PRMEVL(TEST,STARR(7))                                     COLLECT  480
      C                                                                 COLLECT  481
      C     TEST FOR PARAMETER MATCH                                    COLLECT  482
   75 C                                                                 COLLECT  483
         IF(TEST) GO TO 30                                              COLLECT  484
      C                                                                 COLLECT  485
      C     PARAMETER IS NOT ACCEPTED, TRY NEXT P-M GROUP               COLLECT  486
      C                                                                 COLLECT  487
   80    GO TO 20                                                       COLLECT  488
      C                                                                 COLLECT  489
      C     DOES THIS P-M GROUP BEAD HAVE ANY MULTIPLIERS ?             COLLECT  490
      C     STORE DEPTH COUNT IN STARR(9)                               COLLECT  491
      C     DETERMINE DEPTH OF MULTIPLIERS FOR THIS P-M GROUP AND       COLLECT  492
   85 C     PLACE PTR TO BEAD OF THIS SIZE IN STARR(9)                  COLLECT  493
      C                                                                 COLLECT  494
    30    CONTINUE                                                      COLLECT  495
         CALL DMGET(WHOL4,STARR(6),STARR(8))                            COLLECT  496
         IF(STARR(8) .EQ. 0) GO TO 50                                   COLLECT  497
   90    CALL DMGET(WHOL3,STARR(5),MULPTR)                              COLLECT  498
         CALL DMGTBD(STARR(8),STARR(9))                                 COLLECT  499
         IF(STARR(9) .EQ. 0) GO TO 70                                   COLLECT  500
      C                                                                 COLLECT  501
      C     SCAN MULTIPLIER FUNCTION LIST OF ITH MULTIPLIER             COLLECT  502
   95 C                                                                 COLLECT  503
         IDEPTH=1                                                       COLLECT  504
    35   CONTINUE                                                      COLLECT  505
         CALL FNCTEV(MULPTR,TEST,VALMUL)                                COLLECT  506
      C                                                                 COLLECT  507
  100 C     WAS MULTIPLIER ACCEPTED ?                                   COLLECT  508
      C                                                                 COLLECT  509
         IF(TEST) GO TO 40                                              COLLECT  510
      C                                                                 COLLECT  511
      C     TEST FAILED, RETURN                                         COLLECT  512
  105 C                                                                 COLLECT  513
         GO TO 60                                                       COLLECT  514
      C                                                                 COLLECT  515
      C     STORE SUCCESSFUL MULTIPLIER VALUE IN ITH BEAD POSITION      COLLECT  516
      C                                                                 COLLECT  517
  110    CONTINUE                                                       COLLECT  518
         CALL DMSET(WHOL4+IDEPTH,STARR(9),VALMUL)                       COLLECT  519
      C                                                                 COLLECT  520
      C     INCREASE MULTIPLIER DEPTH BY 1                              COLLECT  521
      C                                                                 COLLECT  522
```

```
      IDEPTH=IDEPTH+1                                                   COLLECT  523
      CALL DMGET(WHOL1,MULPTR,MULPTR)                                   COLLECT  524
      IF(MULPTR .EQ. 0) GO TO 50                                        COLLECT  525
      GO TO 35                                                          COLLECT  526
C                                                                       COLLECT  527
C     MULTIPLIER SCAN COMPLETED                                         COLLECT  528
C                                                                       COLLECT  529
 50   CONTINUE                                                          COLLECT  530
      CALL PLOT80                                                       COLLECT  531
C                                                                       COLLECT  532
C     HANDLING OF THIS P-M GROUP COMPLETED. FREE MULTIPLIER             COLLECT  533
C     VALUE BEAD AND CONTINUE P-M GROUP SCAN                            COLLECT  534
C                                                                       COLLECT  535
 60   CONTINUE                                                          COLLECT  536
C                                                                       COLLECT  537
C        FREE PARAMETER MATCH BEADS                                     COLLECT  538
      IF (STARR(7) .EQ. 0) GO TO 65                                     COLLECT  539
      CALL DMGET(WHOL1,STARR(7),NEXT)                                   COLLECT  540
      CALL DMRL80(STARR(7))                                             COLLECT  541
      STARR(7) = NEXT                                                   COLLECT  542
      GO TO 60                                                          COLLECT  543
 65   CONTINUE                                                          COLLECT  544
      IF(STARR(8) .EQ. 0) GO TO 20                                      COLLECT  545
      CALL DMRL80(STARR(9))                                             COLLECT  546
      GO TO 20                                                          COLLECT  547
C                                                                       COLLECT  548
C     NO MORE ROOM IN FILE **ERROR**                                    COLLECT  549
C                                                                       COLLECT  550
 70   CONTINUE                                                          COLLECT  551
      CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,7HPM GRUP)           COLLECT  552
      END                                                               COLLECT  553
```

```
      SUBROUTINE PRMEVL(TEST,IPRMPT)                                    COLLECT  554
      LOGICAL TEST,RESULT                                               COLLECT  555
      INTEGER       WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50   COLLECT  556
      COMMON /COMPCOO/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT  557
C                                                                       COLLECT  558
C     SUBROUTINE EVALUATES PARAMETERS AND BUILDS LIST CONTAINING        COLLECT  559
C     MATCHES                                                           COLLECT  560
C                                                                       COLLECT  561
C     FORMAT FOR READ IN PARAMETER MATCH LIST                           COLLECT  562
C             * PTR TO NEXT MATCH READ  *                               COLLECT  563
C             * CODE FOR PARAMETER NAME *                               COLLECT  564
C             * VALUE OF PARAMETER      *                               COLLECT  565
C             ***************************                               COLLECT  566
C                                                                       COLLECT  567
C     TEST IS SET TO TRUE IF ONE OR MORE PARAMETERS MATCH THIS RECORD   COLLECT  568
C          OTHERWISE FALSE                                              COLLECT  569
C     IPRMPT CONTAINS POINTER TO PARAMETER LIST ON ENTRY                COLLECT  570
C          ON RETURN IPRMPT CONTAINS POINTER TO LIST OF PARAMETER       COLLECT  571
C          MATCHES OR 0 IF NO MATCHES                                   COLLECT  572
C                                                                       COLLECT  573
      IHEAD=LISTPT=0                                                    COLLECT  574
      IPTR=IPRMPT                                                       COLLECT  575
      TEST = .FALSE.                                                    COLLECT  576
C                                                                       COLLECT  577
C     EVALUATE FUNCTION FOR THIS PARAMETER                              COLLECT  578
C                                                                       COLLECT  579
   10 CONTINUE                                                          COLLECT  580
      CALL FNCTEV(IPTR,RESULT,VALPAR)                                   COLLECT  581
      IF (.NOT.RESULT) GO TO 20                                         COLLECT  582
      TEST = .TRUE.                                                     COLLECT  583
      GO TO 30                                                          COLLECT  584
C                                                                       COLLECT  585
C     MATCH FAILED, TRY NEXT PARAMETER                                  COLLECT  586
C                                                                       COLLECT  587
   20 CONTINUE                                                          COLLECT  588
      CALL DMGET(WHOL1,IPTR,IPTR)                                       COLLECT  589
      IF(IPTR .NE. 0) GO TO 10                                          COLLECT  590
      IPRMPT=IHEAD                                                      COLLECT  591
      RETURN                                                            COLLECT  592
C                                                                       COLLECT  593
C     STORE MATCHED PARAMETER IN LIST                                   COLLECT  594
C                                                                       COLLECT  595
   30 CONTINUE                                                          COLLECT  596
      CALL DMGT60(3,NEXT)                                               COLLECT  597
      IF(NEXT .EQ. 0) GO TO 501                                         COLLECT  598
      CALL DMGET(WHOL2,IPTR,NAMPAR)                                     COLLECT  599
      CALL DMSET(WHOL2,NEXT,NAMPAR)                                     COLLECT  600
      CALL DMSET(WHOL3,NEXT,VALPAR)                                     COLLECT  601
C                                                                       COLLECT  602
C     CHECK FOR FIRST READ IN LIST                                      COLLECT  603
C                                                                       COLLECT  604
      IF(LISTPT .NE. 0) GO TO 40                                        COLLECT  605
      IHEAD=LISTPT=NEXT                                                 COLLECT  606
      GO TO 20                                                          COLLECT  607
   40 CONTINUE                                                          COLLECT  608
      CALL DMSET(WHOL1,LISTPT,NEXT)                                     COLLECT  609
      LISTPT=NEXT                                                       COLLECT  610
```

```
                GO TO 20                                                    COLLECT   611
          C                                                                 COLLECT   612
60        C     NO MORE ROOM IN FILE  ** ERROR**                            COLLECT   613
          C                                                                 COLLECT   614
            501 CONTINUE                                                    COLLECT   615
                CALL ERROR1M(26H(21H BEAD SPACE EXHAUSTED),-3,6HPRMEVL)     COLLECT   616
                END                                                         COLLECT   617
```

```
      SUBROUTINE PRMSRH(IPLOTBD)                                        MODA    212
      INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50   COLLECT  619
      COMMON /COMPCOD/ WHOLC,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT 620
      COMMON /STATUS/ STARR(10)                                         COLLECT  621
C                                                                       COLLECT  622
C     THIS SUBROUTINE SEARCHES THE CURVE BEADS WHICH ARE ATTACHED TO    MODA    213
C     THE PLOT BEAD (IPLOTBD) FOR A CURVE BEAD WHOSE PARAMETER NAME     MODA    214
C     AND VALUE MATCH THE PARAMETER NAME AND VALUE FROM THE CURRENT     MODA    215
C     INPUT RECORD. IF A MATCH IS FOUND, THEN THE POINT SPECIFIED BY    MODA    216
C     THE AXIS CARD IS ALLOCATED TO THE MATCHING CURVE BEAD.            MODA    217
C     OTHERWISE, A NEW CURVE BEAD IS ALLOCATED AND THE POINT IS         MODA    218
C     ASSIGNED TO THE NEW CURVE BEAD. POINTS FROM A SINGLE INPUT        MODA    219
C     RECORD MAY BE ASSIGNED TO MORE THAN ONE CURVE BEAD.               MODA    220
C                                                                       MODA    221
C     IF "IPLOTBD" IS ZERO, THEN THIS ROUTINE ALLOCATES A PLOT BEAD     MODA    222
C     AND A CURVE BEAD.                                                 MODA    223
C                                                                       MODA    224
C     STARR(7) CONTAINS THE POINTER TO THE FIRST PARAMETER MATCH        MODA    225
C     BEAD FOR THE CURRENT INPUT RECORD.                                MODA    226
C     STARR(7) CONTAINS POINTER TO LIST OF PARAMETER MATCHES            COLLECT  632
C                                                                       COLLECT  633
C     CURVE BEAD FORMAT                                                 COLLECT  634
C             * NEXT                          *                       COLLECT  635
C             * PARAMETER VALUE               *                       COLLECT  636
C             * COUNT OF POINTS               *                       COLLECT  637
C             * PLOT SYMBOL ID USED BY PS 4    *                      COLLECT  638
C             * CODE FOR PARAMETER NAME       *                       COLLECT  639
C             * RECORD NUMBER POINTER         *                       COLLECT  640
C             * X POINTER                     *                       COLLECT  641
C             * Y1 POINTER                    *                       COLLECT  642
C             * Y2 POINTER                    *                       COLLECT  643
C             * Y3 POINTER                    *                       COLLECT  644
C             *********************************                       COLLECT  645
C                                                                       COLLECT  646
C     FORMAT FOR BEAD IN PARAMETER MATCH LIST                           COLLECT  647
C             * PTR TO NEXT MATCH BEAD  *                              COLLECT  648
C             * CODE FOR PARAMETER NAME *                              COLLECT  649
C             * VALUE OF PARAMETER      *                              COLLECT  650
C             ***************************                             COLLECT  651
C                                                                       COLLECT  652
C     INITIALIZE LOOP FOR PARAMETER VALUES, CHECK FOR NO PARAMETERS     COLLECT  653
C                                                                       COLLECT  654
      IF(STARR(7) .NE. 0) GO TO 5                                       COLLECT  655
      VALPAR=NAMPAR=0                                                   COLLECT  656
      IPLOT = IPLOTBD                                                   MODA    227
      GO TO 10                                                          COLLECT  657
C                                                                       COLLECT  658
C     GET NAME AND VALUE OF THIS PARAMETER                              COLLECT  659
    5 CONTINUE                                                          COLLECT  660
      CALL DMGET(WHOL2,STARR(7),NAMPAR)                                 COLLECT  661
      CALL DMGET(WHOL3,STARR(7),VALPAR)                                 COLLECT  662
      IPLOT = IPLOTBD                                                   MODA    228
   10 NEXT = IPLOT                                                      MODA    229
      IF(NEXT .NE. 0) GO TO 15                                          MODA    230
C                                                                       COLLECT  673
C        NO PLOT BEAD, CREATE ONE                                       COLLECT  674
      CALL GMGTBD(12+2*STARR(9),NEXT)                                   MODA    231
```

```
                      IF (NEXT .EQ. 0) GO TO 501                                COLLECT   676
                      IPLOTBD = NEXT                                            MODA      232
   60           C                                                               MODA      233
                C         ATTACH THE NEW PLOT BEAD TO THE P-M BEAD.              MODA      234
                      CALL DMSET(WHOL5,STARR(6),NEXT)                           MODA      235
                      IPLOT = NEXT                                              MODA      236
                C                                                               COLLECT   680
   65           C        UPDATE OR INITIALIZE LABELS                            COLLECT   681
                   15 CALL LBLSET(NEXT)                                         COLLECT   682
                C                                                               COLLECT   685
                C        STEP ALONG CURVE BEAD LIST                             COLLECT   686
                      ICCMP = WHOL0 + 11                                        MODA      237
   70                 CONTINUE                                                 COLLECT   687
                      CALL DMGET(ICCMP,IPLOT,NEXT)                              COLLECT   688
                      IF(NEXT .NE. 0) GO TO 30                                  COLLECT   689
                C                                                               COLLECT   690
                C     NEW BEAD REQUIRED                                         COLLECT   691
   75           C                                                               COLLECT   692
                      CALL DMGTBD(12,NEXT)                                      MODA      238
                      IF(NEXT .EQ. 0) GO TO 501                                 COLLECT   694
                      CALL DMSET(ICCMP,IPLOT,NEXT)                              COLLECT   695
                      IPLOT=NEXT                                                COLLECT   696
   80                 CALL DMSET(WHOL2,IPLOT,VALPAR)                           COLLECT   697
                      CALL DMSET(WHOL5,IPLOT,NAMPAR)                            COLLECT   698
                      GO TO 40                                                  COLLECT   699
                C                                                               COLLECT   700
                C         MATCH THE PARAMETER NAME AND VALUE WITH A CURVE BEAD. MODA      239
   85           C                                                               COLLECT   702
                   30 CONTINUE                                                  COLLECT   703
                      ICCMP=WHOL1                                               COLLECT   704
                      IPLOT=NEXT                                                COLLECT   705
                      CALL DMGET(WHOL5,IPLOT,NAMPLT)                            COLLECT   706
   90                 IF(NAMPAR .NE. NAMPLT) GO TO 20                          COLLECT   707
                      CALL DMGET(WHOL2,IPLOT,VALPLT)                            COLLECT   708
                      IF(VALPAR .NE. VALPLT) GO TO 20                           COLLECT   709
                C                                                               COLLECT   710
                C         STORE X-Y VALUES IN THE COORDINATE BEADS.             MODA      240
   95           C                                                               COLLECT   712
                   40 CONTINUE                                                  COLLECT   713
                      CALL STORE(IPLOT)                                         COLLECT   714
                C                                                               COLLECT   715
                C        STEP TO OBTAIN THE NEXT PARAMETER MATCH BEAD.          MODA      241
  100           C                                                               COLLECT   717
                      IF(STARR(7) .EQ. 0) RETURN                                COLLECT   718
                      CALL DMGET(WHOL1,STARR(7),NEXT)                           COLLECT   719
                      CALL DMRLBD(STARR(7))                                     COLLECT   720
                      IF(NEXT .EQ. 0) RETURN                                    COLLECT   721
  105                 STARR(7)=NEXT                                             COLLECT   722
                      GO TO 5                                                   COLLECT   723
                C                                                               COLLECT   724
                C     NO CORE ROOM IN FILE  **ERROR**                           COLLECT   725
                C                                                               COLLECT   726
  110              501 CONTINUE                                                 COLLECT   727
                      CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,6HPRMSRH)    COLLECT   728
                      END                                                       COLLECT   729
```

```
      PROGRAM PS3MAIN                                                    COLLECT  730
      LOGICAL LSTELE,ROCHK                                               COLLECT  731
      INTEGER FMT(5)                                                     COLLECT  732
      INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50    COLLECT  733
      INTEGER TAPENO,UNITNO,ELETAB,FUNCTB,SURFTB                         COLLECT  734
      COMMON /HEADPTR/ IHEADR                                            COLLECT  735
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                          COLLECT  736
      COMMON /STATUS/ STARR(10)                                          COLLECT  737
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT  738
      COMMON /INPTRC/ V(100),NOOFRD,NOOFRC,ROFMT(24)                     COLLECT  739
      COMMON /RTBDSZ/ NOFPTS                                             COLLECT  740
      COMMON /TAPE/ TAPENO,UNITNO                                        COLLECT  741
      COMMON /ERROR/ ROCHK                                               COLLECT  742
      COMMON /SFTAB/ SURFTB(10,11)                                       COLLECT  743
      DATA ID /7HPS3MAIN/                                                COLLECT  744
C                                                                        COLLECT  745
C     SUBROUTINE CONTROLS OVERALL SCAN OF TAPE, READS EACH RECORD        COLLECT  746
C     AND RETURNS TO MAIN WHEN SCAN HAS COMPLETED                        COLLECT  747
C                                                                        COLLECT  748
C     STATUS ARRAY FORMAT INTEGER ARRAY STARR(10)                        COLLECT  749
C                                                                        COLLECT  750
C           * STARR(1)= NUMBER OF CURRENT TAPE                           COLLECT  751
C           * STARR(2)= PTR TO CURRENT PLOT GROUP BEAD                   COLLECT  752
C           * STARR(3)= PTR TO CURRENT CONDITION BEAD                    COLLECT  753
C           * STARR(4)= PTR TO CURRENT FUNCTION BEAD                     COLLECT  754
C           * STARR(5)= PTR TO CURRENT AXIS BEAD                         COLLECT  755
C           * STARR(6)= PTR TO CURRENT P-M GROUP BEAD                    COLLECT  756
C           * STARR(7)= VALUE OF SUCCESSFUL PARAMETER MATCH              COLLECT  757
C           * STARR(8)= VALUE OF MULTIPLIER DEPTH COUNTER                COLLECT  758
C           * STARR(9)= PTR TO BEAD CONTAINING VALUES OF SUCCESSFUL      COLLECT  759
C                      MULTIPLIER MATCHES                                COLLECT  760
C           * STARR(10)= NUMBER OF Y AXES                                COLLECT  761
C                                                                        COLLECT  762
C                                                                        COLLECT  763
C     CHECK FOR NULL DATA HANDLER FILE                                   COLLECT  764
C                                                                        COLLECT  765
      IF(IHEADR .EQ. 0) GO TO 515                                        COLLECT  766
      NOOFRC=0                                                           COLLECT  767
      STARR(1)=TAPENO                                                    COLLECT  768
C                                                                        COLLECT  769
C     GET NEXT RECORD                                                    COLLECT  770
C                                                                        COLLECT  771
   10 CONTINUE                                                           COLLECT  772
      NOOFRC=NOOFRC+1                                                    COLLECT  773
      CALL READRC(LSTELE),RETURNS(512)                                   COLLECT  774
C                                                                        COLLECT  775
C     TEST FOR LAST ELEMENT                                              COLLECT  776
C                                                                        COLLECT  777
      IF(LSTELE) GO TO 20                                                COLLECT  778
      STARR(2)=IHEADR                                                    COLLECT  779
C                                                                        COLLECT  780
C     BEGIN SCAN OF PLOT GROUPS                                          COLLECT  781
      CALL SCANPG                                                        COLLECT  782
C                                                                        COLLECT  783
C     ALL SCANS FOR THIS RECORD COMPLETE                                 COLLECT  784
C                                                                        COLLECT  785
      GO TO 10                                                           COLLECT  786
```

```
      C                                                                 COLLECT   787
      C     LAST ELEMENT, RETURN TO MAIN OVERLAY FOR PLOTTING           COLLECT   788
60    C                                                                 COLLECT   789
        20  CONTINUE                                                    COLLECT   790
            GO TO 600                                                   COLLECT   791
      C                                                                 COLLECT   792
      C     ERROR DURING READ OF CURRENT RECORD, RECORD DELETED         COLLECT   793
65    C                                                                 COLLECT   794
       512  CONTINUE                                                    COLLECT   795
            ENCODE(42,1000,FMT) NOOFRC                                  COLLECT   796
            CALL ERROR#(FMT,3,ID)                                       COLLECT   797
            GO TO 10                                                    COLLECT   798
70     515  CALL ERROR#(33H(28H NO DATA HANDLER FILE EXISTS),-3,ID)     COLLECT   799
       600  CONTINUE                                                    COLLECT   800
      1000  FORMAT(37H(37H ERROR DURING READ OF RECORD ,I7,2H ))        COLLECT   801
            END                                                         COLLECT   802
```

```
      FUNCTION RCOVAL(NAME)                                          COLLECT   803
      COMMON /INPTRC/ V(100),NOOFWD,NOOFRC,RDFMT(24)                 COLLECT   804
C                                                                    COLLECT   805
C                                                                    COLLECT   806
      IF(NAME .LT. 1 .OR. NAME .GT. NOOFWD) GO TO 511                COLLECT   807
      RCOVAL=V(NAME)                                                 COLLECT   808
      RETURN                                                         COLLECT   809
C                                                                    COLLECT   810
C     ATTEMPT TO REQUEST INVALID RECORD ELEMENT                      COLLECT   811
C                                                                    COLLECT   812
  511 CALL ERROR(17H(12H INV ELEMENT),-3,6HRCOVAL)                   COLLECT   813
      END                                                            COLLECT   814
```

```
      SUBROUTINE READRC(LSTELE),RETURNS(ERROR)                          COLLECT   815
      LOGICAL LSTELE,RDCHK                                              COLLECT   816
      INTEGER TAPENO,UNITNO                                             COLLECT   817
      COMMON /ERROR/ RDCHK                                              COLLECT   818
      COMMON /INPTRC/ V(100),NOOFWD,NOOFRC,RDFMT(24)                    COLLECT   819
      COMMON /TAPE/ TAPENO,UNITNO                                       COLLECT   820
C                                                                       COLLECT   821
C                                                                       COLLECT   822
C     SUBROUTINE GETS RECORD FROM TAPE AND STORES THE ELEMENT VALUES    COLLECT   823
C     INTO THE V ARRAY.  THIS ARRAY IS AVAILABLE TO INITL AND SPCVAL    COLLECT   824
C     AND RDVAL IN COMMON                                               COLLECT   825
C                                                                       COLLECT   826
      RDCHK=.FALSE.                                                     COLLECT   827
      READ(UNITNO,RDFMT) (V(I),I=1,NOOFWD)                              COLLECT   828
      IF(RDCHK) GO TO 30                                                COLLECT   829
C                                                                       COLLECT   830
C     CHECK FOR LAST RECORD                                             COLLECT   831
C                                                                       COLLECT   832
      IF(ECF(UNITNO) .NE. 0.) GO TO 20                                  COLLECT   833
      DO 10 I=1,NOOFWD                                                  COLLECT   834
      IF(V(I) .NE. -9.) GO TO 15                                        COLLECT   835
   10 CONTINUE                                                          COLLECT   836
      GO TO 20                                                          COLLECT   837
   15 CONTINUE                                                          COLLECT   838
      LSTELE=.FALSE.                                                    COLLECT   839
      RETURN                                                            COLLECT   840
   20 CONTINUE                                                          COLLECT   841
      LSTELE=.TRUE.                                                     COLLECT   842
      RETURN                                                            COLLECT   843
C                                                                       COLLECT   844
C     ERROR IN READING TAPE, THIS RECORD DROPPED                        COLLECT   845
C                                                                       COLLECT   846
   30 CONTINUE                                                          COLLECT   847
      RETURN ERROR                                                      COLLECT   848
      END                                                               COLLECT   849
```

```
      SUBROUTINE SCANCN(TEST)                                                  COLLECT  850
      LOGICAL TEST                                                             COLLECT  851
      INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50          COLLECT  852
      COMMON /STATUS/ STARR(10)                                                COLLECT  853
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50       COLLECT  854
C                                                                              COLLECT  855
C     SUBROUTINE SCANS CONDITION LIST                                          COLLECT  856
C                                                                              COLLECT  857
C     TEST IS USED TO INDICATE WHETHER THE CURRENT RECORD HAS PASSED           COLLECT  858
C     THE SET OF CONDITIONS                                                    COLLECT  859
C                                                                              COLLECT  860
C     CONDITION READ FORMAT                                                    COLLECT  861
C             * NEXT CONDITION ELEMENT    *                                  COLLECT  862
C             * CODED ELEMENT NAME         *                                  COLLECT  863
C             * PTR TO LIST OF FUNCTIONS *                                    COLLECT  864
C             ****************************                                     COLLECT  865
C                                                                              COLLECT  866
C     SET UP FIRST CONDITION ELEMENT NAME                                      COLLECT  867
C                                                                              COLLECT  868
      CALL DMGET(WHOL3,STARR(2),STARR(3))                                      COLLECT  869
      TEST=.TRUE.                                                              COLLECT  870
C                                                                              COLLECT  871
C     CHECK FOR LAST CONDITION                                                 COLLECT  872
C                                                                              COLLECT  873
   10 IF(STARR(3) .EQ. 0) RETURN                                               COLLECT  874
C                                                                              COLLECT  875
C     EVALUATE ALL FUNCTIONS ASSOCIATED WITH THIS CONDITION                    COLLECT  876
C                                                                              COLLECT  877
      CALL FNCTEV(STARR(2),TEST,VAL)                                           COLLECT  878
C                                                                              COLLECT  879
C     IF THIS CONDITION FAILED, RETURN WITH TEST = FALSE                       COLLECT  880
C                                                                              COLLECT  881
      IF(TEST) GO TO 20                                                        COLLECT  882
      RETURN                                                                   COLLECT  883
C                                                                              COLLECT  884
C     CURRENT CONDITION PASSED, STEP TO NEXT CONDITION                         COLLECT  885
C                                                                              COLLECT  886
   20 CALL DMGET(WHOL1,STARR(3),STARR(3))                                      COLLECT  887
      GO TO 10                                                                 COLLECT  888
      END                                                                      COLLECT  889
```

```
      SUBROUTINE SCANPG                                                 COLLECT   890
      LOGICAL TEST                                                      COLLECT   891
      INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50   COLLECT   892
      COMMON /STATUS/ STARR(10)                                         COLLECT   893
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50 COLLECT  894
C                                                                       COLLECT   895
C     SUBROUTINE CONDUCTS SCAN OF PLOT GROUP BEADS                      COLLECT   896
C                                                                       COLLECT   897
C     1ST PLOT GROUP BEAD POINTER STORED IN STARR(2) BY SCANMN          COLLECT   898
C                                                                       COLLECT   899
C                                                                       COLLECT   900
C     PLOT GROUP BEAD FORMAT                                            COLLECT   901
C           * PTR TO NEXT PLOT GROUP BEAD *                             COLLECT   902
C           * PTR TO TITLE BEAD LIST      *                            COLLECT   903
C           * PTR TO CONDITION LIST       *                            COLLECT   904
C           * PTR TO AXES LIST            *                            COLLECT   905
C           * PTR TO CONDITION TEXT LIST  *                            COLLECT   906
C           * NO OF CONDITION TEXT BEADS  *                            COLLECT   907
C           ******************************                            COLLECT   908
C                                                                       COLLECT   909
C     BEGIN CONDITION SCAN                                              COLLECT   910
C                                                                       COLLECT   911
   10 CALL SCANCN(TEST)                                                 COLLECT   912
C                                                                       COLLECT   913
C     IF CONDITIONS WERE SATISFIED BEGIN P M GROUP SCAN, OTHERWISE      COLLECT   914
C     GET NEXT PLOT GROUP                                               COLLECT   915
C                                                                       COLLECT   916
      IF(TEST)CALL PM GRUP                                              COLLECT   917
C                                                                       COLLECT   918
C     SCAN FOR THIS PLOT GROUP COMPLETE, PLACE NEXT PLOT GROUP BEAD     COLLECT   919
C     PTR IN STARR(2)                                                   COLLECT   920
C                                                                       COLLECT   921
      CALL DMGET(WHOL1,STARR(2),STARR(2))                               COLLECT   922
C                                                                       COLLECT   923
C     TEST FOR LAST PLOT GROUP                                          COLLECT   924
C                                                                       COLLECT   925
      IF(STARR(2) .EQ. 0) RETURN                                        COLLECT   926
      GO TO 10                                                          COLLECT   927
      END                                                               COLLECT   928
```

```
      FUNCTION SPCVAL(NAME)                                                  COLLECT   929
      INTEGER ELETAB,FUNCTB,SURFTB,ZTRUNC,FMT1(7)                            COLLECT   930
      DIMENSION ISURF(10),SURF(10)                                           COLLECT   931
      EQUIVALENCE(ISURF,SURF)                                                COLLECT   932
      COMMON /ITABLE/ ELETAB,FUNCTB,XSCAL,YSCAL                              COLLECT   933
      COMMON /SETAB/ SURFTB(10,11)                                           COLLECT   934
      COMMON /CONSTN/ BOVERC,NOOFDG                                          COLLECT   935
      COMMON /INPTRC/ V(100),NOOFNO,NOOFRC,RDFMT(24)                         COLLECT   936
      DATA NOTSET /1H /                                                      COLLECT   937
      DATA RADDG /.0174533293/                                               COLLECT   938
      DATA ID/6HSPCVAL/                                                      COLLECT   939
C                                                                            COLLECT   940
C     SUBROUTINE EVALUATES SPECIAL ELEMENTS USING APPROPRIATE FUNCTIONS      COLLECT   941
C                                                                            COLLECT   942
C     SPECIAL ELEMENTS ARE ASSIGNED CODES 501 THRU 599                       COLLECT   943
C                                                                            COLLECT   944
C     SURFACES ARE ASSIGNED CODES 601 THRU 699                               COLLECT   945
C                                                                            COLLECT   946
C                                                                            COLLECT   947
C     GET SURFACE CODE DIGITS FROM V4 FOR USE BY FUNCTIONS                   COLLECT   948
C     NOTE V4 MUST BE PROPERLY ROUNDED                                       COLLECT   949
C                                                                            COLLECT   950
      IV4=V(4)+.5                                                            COLLECT   951
      ENCODE(10,1000,IV4) IV4                                                COLLECT   952
      DO  200 I=1,10                                                         COLLECT   953
      ISURF(I)=NOTSET                                                        COLLECT   954
  200 CONTINUE                                                               COLLECT   955
      DO 210 I=1,NOOFDG                                                      COLLECT   956
      ISURF(I)=ZTRUNC(IV4,11-I)                                              COLLECT   957
      ENCODE(10,1001,ISURF(I))ISURF(I)                                       COLLECT   958
      IF(ISURF(I) .EQ. 0) ISURF(I)=10                                        COLLECT   959
  210 CONTINUE                                                               COLLECT   960
      DO 220 I=1,NOOFDG                                                      COLLECT   961
      IF(SURFTB(I,11) .EQ. NOTSET) GO TO 215                                 COLLECT   962
      ISURF(I)=SURFTB(I,ISURF(SURFTB(I,11)))                                 COLLECT   963
      GO TO 220                                                              COLLECT   964
  215 CONTINUE                                                               COLLECT   965
      ISURF(I)=NOTSET                                                        COLLECT   966
  220 CONTINUE                                                               COLLECT   967
C                                                                            COLLECT   968
C     DETERMINE TYPE OF SPECIAL ELEMENT - FUNCTION OR SURFACE                COLLECT   969
C                                                                            COLLECT   970
      IF(NAME .GT. 500 .AND. NAME .LE. 599) GO TO 500                        COLLECT   971
      IF(NAME .GT. 600 .AND. NAME .LE. 699) GO TO 600                        COLLECT   972
      GO TO 800                                                              COLLECT   973
C                                                                            COLLECT   974
C     A TOTAL OF 20 ELEMENTS AND 10 SURFACES ARE AVAILABLE                   COLLECT   975
C     FOR USE BY THE FUNCTIONS                                               COLLECT   976
C     REFERENCES TO ELEMENTS SHOULD BE MADE AS V(I)  I=1,2,...,20            COLLECT   977
C     REFERENCES TO SURFACES SHOULD BE MADE AS SURF(I)  I=1,2,...,10         COLLECT   978
C     THE PROGRAM SHOULD CHECK FOR AN UNDEFINED SURFACE CODE                 COLLECT   979
C     ISURF(I) .NE. NOTSET                                                   COLLECT   980
C                                                                            COLLECT   981
C                                                                            COLLECT   982
C     NAME IS A FUNCTION                                                     COLLECT   983
C                                                                            COLLECT   984
  500 CONTINUE                                                               COLLECT   985
```

```
                TEMP=-V(8)*RADPDG                                                  COLLECT   986
                COSV8=COS(TEMP)                                                    COLLECT   987
60              SINV8=SIN(TEMP)                                                    COLLECT   988
                TEMP=V(9)*RADPDG                                                   COLLECT   989
                COSV9=COS(TEMP)                                                    COLLECT   990
                SINV9=SIN(TEMP)                                                    COLLECT   991
                NAME=NAME-500                                                      COLLECT   992
65              IF (NAME .GT. 35) GO TO 800                                        COLLECT   993
                GO TO (501,502,503,504,505,506,507,508,509,510,511,512,513,514,    COLLECT   994
               X 515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,      COLLECT   995
               X 530,531,532,533,534,535),NAME                                     COLLECT   996
          C                                                                        COLLECT   997
70        C       C1                                                             COLLECT   998
            501 CONTINUE                                                         COLLECT   999
                SPCVAL=-V(11)*COSV9*SINV8+V(12)*COSV8-V(10)*SINV9*SINV8         COLLECT  1000
                RETURN                                                           COLLECT  1001
          C                                                                      COLLECT  1002
75        C       C2                                                             COLLECT  1003
            502 CONTINUE                                                         COLLECT  1004
                SPCVAL= V(14)*BOVERC*COSV9*SINV8+V(13)*COSV8+V(15)*BOVERC*SINV9  COLLECT  1005
               X *SINV8                                                          COLLECT  1006
                RETURN                                                           COLLECT  1007
80        C                                                                      COLLECT  1008
          C       C3                                                             COLLECT  1009
            503 CONTINUE                                                         COLLECT  1010
                SPCVAL= V(14)*COSV9*COSV8-(V(13)*SINV8)/BOVERC+V(15)*SINV9*COSV8  COLLECT  1011
                RETURN                                                           COLLECT  1012
85        C                                                                      COLLECT  1013
          C       C4                                                             COLLECT  1014
            504 CONTINUE                                                         COLLECT  1015
                SPCVAL=-V(14)*SINV9+V(15)*COSV9                                  COLLECT  1016
                RETURN                                                           COLLECT  1017
90        C                                                                      COLLECT  1018
          C       C5                                                             COLLECT  1019
            505 CONTINUE                                                         COLLECT  1020
                SPCVAL=V(14)*COSV9+V(15)*SINV9                                   COLLECT  1021
                RETURN                                                           COLLECT  1022
95        C                                                                      COLLECT  1023
          C       C6                                                             COLLECT  1024
            506 CONTINUE                                                         COLLECT  1025
                SPCVAL=V(11)*COSV9+V(10)*SINV9                                   COLLECT  1026
                RETURN                                                           COLLECT  1027
100       C                                                                      COLLECT  1028
          C       C7                                                             COLLECT  1029
            507 CONTINUE                                                         COLLECT  1030
                SPCVAL=V(10)*COSV9-V(11)*SINV9                                   COLLECT  1031
                RETURN                                                           COLLECT  1032
105       C                                                                      COLLECT  1033
          C       C8                                                             COLLECT  1034
            508 CONTINUE                                                         COLLECT  1035
                SPCVAL=V(11)*COSV9*COSV8+V(12)*(-SINV8)+V(10)*SINV9*COSV8        COLLECT  1036
                RETURN                                                           COLLECT  1037
110       C                                                                      COLLECT  1038
          C       C9                                                             COLLECT  1039
            509 CONTINUE                                                         COLLECT  1040
                SPCVAL=(V(10)*COSV9-V(11)*SINV9)/(V(11)*COSV9*COSV8+V(12)*(-SINV8)  COLLECT  1041
               X +V(10)*SINV9*COSV8)                                             COLLECT  1042
```

```
      RETURN
C
C        C10
  510 CONTINUE
      SPCVAL=V(5)*(V(10)*COSV9-V(11)*SINV9)/(V(11)*COSV9*COSV8+V(12)
     X *(-SINV8)+V(13)*SINV9*COSV8)
      RETURN
C
C        C11
  511 CONTINUE
      GO TO 830
      RETURN
C
C        C12
  512 CONTINUE
      GO TO 830
      RETURN
C
C        C13
  513 CONTINUE
      GO TO 830
      RETURN
C
C        C14
  514 CONTINUE
      GO TO 830
      RETURN
C
C        C15
  515 CONTINUE
      GO TO 830
      RETURN
C
C        FUN1
  516 CONTINUE
      GO TO 830
      RETURN
C
C        FUN2
  517 CONTINUE
      GO TO 830
      RETURN
C
C        FUN3
  518 CONTINUE
      GO TO 830
      RETURN
C
C        FUN4
  519 CONTINUE
      GO TO 830
      RETURN
C
C        FUN5
  520 CONTINUE
      GO TO 830
      RETURN
```

```
COLLECT 1043
COLLECT 1044
COLLECT 1045
COLLECT 1046
COLLECT 1047
COLLECT 1048
COLLECT 1049
COLLECT 1050
COLLECT 1051
COLLECT 1052
COLLECT 1053
COLLECT 1054
COLLECT 1055
COLLECT 1056
COLLECT 1057
COLLECT 1058
COLLECT 1059
COLLECT 1060
COLLECT 1061
COLLECT 1062
COLLECT 1063
COLLECT 1064
COLLECT 1065
COLLECT 1066
COLLECT 1067
COLLECT 1068
COLLECT 1069
COLLECT 1070
COLLECT 1071
COLLECT 1072
COLLECT 1073
COLLECT 1074
COLLECT 1075
COLLECT 1076
COLLECT 1077
COLLECT 1078
COLLECT 1079
COLLECT 1080
COLLECT 1081
COLLECT 1082
COLLECT 1083
COLLECT 1084
COLLECT 1085
COLLECT 1086
COLLECT 1087
COLLECT 1088
COLLECT 1089
COLLECT 1090
COLLECT 1091
COLLECT 1092
COLLECT 1093
COLLECT 1094
COLLECT 1095
COLLECT 1096
COLLECT 1097
COLLECT 1098
COLLECT 1099
```

```
                C                                                       COLLECT 1100
                C       FUN6                                            COLLECT 1101
                  521 CONTINUE                                          COLLECT 1102
175                     GO TO 830                                       COLLECT 1103
                        RETURN                                          COLLECT 1104
                C                                                       COLLECT 1105
                C       FUN7                                            COLLECT 1106
                  522 CONTINUE                                          COLLECT 1107
180                     GO TO 830                                       COLLECT 1108
                        RETURN                                          COLLECT 1109
                C                                                       COLLECT 1110
                C       FUN8                                            COLLECT 1111
                  523 CONTINUE                                          COLLECT 1112
185                     GO TO 830                                       COLLECT 1113
                        RETURN                                          COLLECT 1114
                C                                                       COLLECT 1115
                C       FUN9                                            COLLECT 1116
                  524 CONTINUE                                          COLLECT 1117
190                     GO TO 830                                       COLLECT 1118
                        RETURN                                          COLLECT 1119
                C                                                       COLLECT 1120
                C       FUN10                                           COLLECT 1121
                  525 CONTINUE                                          COLLECT 1122
195                     GO TO 830                                       COLLECT 1123
                        RETURN                                          COLLECT 1124
                C                                                       COLLECT 1125
                C       FUN11                                           COLLECT 11
                  526 CONTINUE                                          COLLECT 11
200                     GO TO 830                                       COLLECT 1128
                        RETURN                                          COLLECT 1129
                C                                                       COLLECT 1130
                C       FUN12                                           COLLECT 1131
                  527 CONTINUE                                          COLLECT 1132
205                     GO TO 830                                       COLLECT 1133
                        RETURN                                          COLLECT 1134
                C                                                       COLLECT 1135
                C       FUN13                                           COLLECT 1136
                  528 CONTINUE                                          COLLECT 1137
210                     GO TO 830                                       COLLECT 1138
                        RETURN                                          COLLECT 1139
                C                                                       COLLECT 1140
                C       FUN14                                           COLLECT 1141
                  529 CONTINUE                                          COLLECT 1142
215                     GO TO 830                                       COLLECT 1143
                        RETURN                                          COLLECT 1144
                C                                                       COLLECT 1145
                C       FUN15                                           COLLECT 1146
                  530 CONTINUE                                          COLLECT 1147
220                     GO TO 830                                       COLLECT 1148
                        RETURN                                          COLLECT 1149
                C                                                       COLLECT 1150
                C       FUN16                                           COLLECT 1151
                  531 CONTINUE                                          COLLECT 1152
225                     GO TO 830                                       COLLECT 1153
                        RETURN                                          COLLECT 1154
                C                                                       COLLECT 1155
                C       FUN17                                           COLLECT 1156
```

```
      532 CONTINUE                                                      COLLECT 1157
          GO TO 830                                                     COLLECT 1158
          RETURN                                                        COLLECT 1159
C                                                                       COLLECT 1160
C         FUN18                                                         COLLECT 1161
      533 CONTINUE                                                      COLLECT 1162
          GO TO 830                                                     COLLECT 1163
          RETURN                                                        COLLECT 1164
C                                                                       COLLECT 1165
C         FUN19                                                         COLLECT 1166
      534 CONTINUE                                                      COLLECT 1167
          GO TO 830                                                     COLLECT 1168
          RETURN                                                        COLLECT 1169
C                                                                       COLLECT 1170
C         FUN20                                                         COLLECT 1171
      535 CONTINUE                                                      COLLECT 1172
          GO TO 830                                                     COLLECT 1173
          RETURN                                                        COLLECT 1174
C                                                                       COLLECT 1175
C     NAME IS A SURFACE                                                 COLLECT 1176
C                                                                       COLLECT 1177
      600 CONTINUE                                                      COLLECT 1178
          NAME=NAME-500                                                 COLLECT 1179
          IF(NAME .GT. NOOFDG) GO TO 810                                COLLECT 1180
          IF(ISURF(NAME) .EQ. NOTSET) GO TO 820                         COLLECT 1181
          SPCVAL=SURF(NAME)                                             COLLECT 1182
          RETURN                                                        COLLECT 1183
C                                                                       COLLECT 1184
C     ERRORS SPCVAL                                                     COLLECT 1185
C                                                                       COLLECT 1186
      800 CONTINUE                                                      COLLECT 1187
          ENCODE(40,1002,FMT1) NAME                                     COLLECT 1188
          GO TO 850                                                     COLLECT 1189
      810 CONTINUE                                                      COLLECT 1190
          ENCODE(40,1003,FMT1) NAME                                     COLLECT 1191
          GO TO 850                                                     COLLECT 1192
      820 CONTINUE                                                      COLLECT 1193
          ENCODE(40,1004,FMT1) NAME,NOOFRC                              COLLECT 1194
          CALL ERRORM(FMT1,+3,ID)                                       COLLECT 1195
C                                                                       COLLECT 1196
C     IF CODE DIGIT IS NOT SET, ERROR VALUE 1111. IS ASSIGNED           COLLECT 1197
C                                                                       COLLECT 1198
          SPCVAL=1111.                                                  COLLECT 1199
          RETURN                                                        COLLECT 1200
      830 CONTINUE                                                      COLLECT 1201
          ENCODE(40,1005,FMT1) NAME                                     COLLECT 1202
      850 CONTINUE                                                      COLLECT 1203
          CALL ERRORM(FMT1,-3,ID)                                       COLLECT 1204
 1000 FORMAT(I10)                                                       COLLECT 1205
 1001 FORMAT(I1,9X)                                                     COLLECT 1206
 1002 FORMAT(26H(26H INV SPC ELEMENT CODE ,I4,1H))                      COLLECT 1207
 1003 FORMAT(34H(34H SURF REQUESTED NOT IN V4, S= ,I4,1H))              COLLECT 1208
 1004 FORMAT(57H(31H SURF CODE DIGIT NOT SET, S= ,I2,15H RECORD = ,I5,1H COLLECT 1209
     *))                                                                COLLECT 1210
 1005 FORMAT(17H(17H UNDEF FUNCT ,I4,1H))                               COLLECT 1211
          END                                                           COLLECT 1212
```

```
      SUBROUTINE STORE(IHEAD)                                                  COLLECT  1213
C                                                                              COLLECT  1214
      INTEGER STARR,WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50          COLLECT  1215
      COMMON /COMPCOD/ WHOL0,WHOL1,WHOL2,WHOL3,WHOL4,WHOL5,ARRY20,ARRY50       COLLECT  1216
      COMMON /STATUS/ STARR(10)                                                COLLECT  1217
      COMMON /INPTRC/ V(100),NOOFWD,NOOFRC,RDFMT(24)                           COLLECT  1218
      COMMON /PTBUSZ/ NOFPTS                                                   COLLECT  1219
C                                                                              COLLECT  1220
C     SUBROUTINE PLACES X-Y VALUES INTO BEAD                                   COLLECT  1221
C                                                                              COLLECT  1222
C     IHEAD POINTS TO 5 WORD PLOT HEADER BEAD TO WHICH THE X-Y BEADS ARE       COLLECT  1223
C     ATTACHED                                                                 COLLECT  1224
C     POINTS CONTAINS X AND Y VALUES TO BE STORED                              COLLECT  1225
C                                                                              COLLECT  1226
C     POINT HEADER BEAD FORMAT                                                 COLLECT  1227
C             * NEXT                          *                                COLLECT  1228
C             * PARAMETER VALUE               *                                COLLECT  1229
C             * COUNT OF POINTS               *                                COLLECT  1230
C             * PLOT SYMBOL ID USED BY PS 4 *                                  COLLECT  1231
C             * CODE FOR PARAMETER NAME       *                                COLLECT  1232
C             * RECORD NUMBER POINTER         *                                COLLECT  1233
C             * X POINTER                     *                                COLLECT  1234
C             * Y1 POINTER                    *                                COLLECT  1235
C             * Y2 POINTER                    *                                COLLECT  1236
C             *    ...                        *                                COLLECT  1237
C             * Y N POINTER                   *                                COLLECT  1238
C             *********************************                                COLLECT  1239
C                                                                              COLLECT  1240
C     X-Y BEAD FORMAT                                                          COLLECT  1241
C             * VALUE 1  *                                                     COLLECT  1242
C             * VALUE 2  *                                                     COLLECT  1243
C             * VALUE 3  *                                                     COLLECT  1244
C             *    -     *                                                     COLLECT  1245
C             *    -     *                                                     COLLECT  1246
C             *    -     *                                                     COLLECT  1247
C             * NOFPTS   *                                                     COLLECT  1248
C             * NEXT     *                                                     COLLECT  1249
C             ************                                                     COLLECT  1250
C                                                                              COLLECT  1251
C     GET COUNT AND NUMBER OF Y AXES                                           COLLECT  1252
C                                                                              COLLECT  1253
      CALL DMGET(WHOL3,IHEAD,ICOUNT)                                           COLLECT  1254
      J=STARR(10)+2                                                            COLLECT  1255
      I=MOD(ICOUNT,NOFPTS)                                                     COLLECT  1256
      ICOUNT=ICOUNT+1                                                          COLLECT  1257
      CALL DMSET(WHOL3,IHEAD,ICOUNT)                                           COLLECT  1258
C                                                                              COLLECT  1259
C     LOOP TO STORE X-Y VALUES IN BEADS                                        COLLECT  1260
C                                                                              COLLECT  1261
      DO 50 K=1,J                                                              COLLECT  1262
      ICOMP=WHOL5+K                                                            COLLECT  1263
      LOCSO=IHEAD                                                              COLLECT  1264
C                                                                              COLLECT  1265
C     STEP POINTERS TO CORRECT POINT BEAD                                      COLLECT  1266
C                                                                              COLLECT  1267
   10 CALL DMGET(ICOMP,LOCSO,NEXT)                                             COLLECT  1268
      IF(NEXT .EQ. 0) GO TO 20                                                 COLLECT  1269
```

```
                  LOCBC=NEXT                                                    COLLECT 1270
                  ICOMP=WHOL1+NOFPTS                                            COLLECT 1271
   60             GO TO 10                                                      COLLECT 1272
                C                                                               COLLECT 1273
                C     IS THERE ROOM LEFT IN THE BEAD FOR THIS POINT?            COLLECT 1274
                C                                                               COLLECT 1275
                 20   CONTINUE                                                  COLLECT 1276
   65            IF(I .NE. 0) GO TO 40                                         COLLECT 1277
                 CALL DMGTBD(NOFPTS+1,NEWBD)                                   COLLECT 1278
                 IF(NEWBD .EQ. 0) GO TO 60                                     COLLECT 1279
                 CALL DMSET(ICOMP,LOCBC,NEWBD)                                 COLLECT 1280
                 LOCBC=NEWBD                                                   COLLECT 1281
   70          C                                                               COLLECT 1282
               C                                                                COLLECT 1283
                 40   CONTINUE                                                  COLLECT 1284
                 ICOMP=WHOL1+I+1                                               COLLECT 1285
                 IF(K .NE. 1) GO TO 45                                         COLLECT 1286
   75            CALL DMSET(ICOMP,LOCBC,NOOFRC)                                COLLECT 1287
                 GO TO 50                                                      COLLECT 1288
                   45 CONTINUE                                                 COLLECT 1289
                 JCOMP=WHOL2+K                                                 COLLECT 1290
                 IF(K .EQ. 2) JCOMP=WHOL2                                      COLLECT 1291
   80            CALL DMGET(JCOMP,STARR(5),IELENO)                             COLLECT 1292
                 VAL = GETVAL(IELENO)                                          COLLECT 1293
                 CALL DMSET(ICOMP,LOCBC,VAL)                                   COLLECT 1294
                 50   CONTINUE                                                 COLLECT 1295
                 RETURN                                                        COLLECT 12
   85          C                                                               COLLECT 12
               C     FILE IS FILLED  **ERROR**  ABORT                          COLLECT 1298
               C                                                               COLLECT 1299
                   60 CONTINUE                                                 COLLECT 1300
                 CALL ERRDMM(26H(21H BEAD SPACE EXHAUSTED),-3,5HSTORE)         COLLECT 1301
   90            END                                                           COLLECT 1302
```

```
      1                 OVERLAY(F,0)                                                    REFORM      2
                        SUBROUTINE ASSIGN(IPTR)                                         REFORM      3
                  C                                                                     REFORM      4
                  C       THIS SUBROUTINE ASSIGNS SYMBOLS TO THE VARIOUS POINT          REFORM      5
      5           C       BEADS AT THE PLOT BEAD LEVEL SO THAT EACH SET OF POINTS IN THE REFORM      6
                  C       SAME PLOT WILL BE UNIQUELY DISPLAYED                          REFORM      7
                  C                                                                     REFORM      8
                        COMMON/CVECTOR/ IPGPT, ICONPT, IAXSPT, IPMPT                    REFORM      9
                        COMMON/CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5, REFORM     10
     10                X IWHOLE6                                                       REFORM     11
                        COMMON/AXESNMS/ NAMEX,NAMEY(3),NOFYAX,IX,IY(3)                  REFORM     12
                        COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                       REFORM     13
                        LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                              REFORM     14
                        MAX = 1                                                         REFORM     15
     15           C                                                                     REFORM     16
                  C       CHECK IF SEQUENTIAL ASSIGNMENT IS TO BE DONE                  REFORM     17
                        IF (.NOT. ISTATP) GO TO 75                                      REFORM     18
                  C                                                                     REFORM     19
                  C       GET POINTER TO PARAMETER LIST BEAD                            REFORM     20
     20                 CALL DMGET(IWHOLE3,IPMPT,INPTR)                                MODA      242
                  C                                                                     REFORM     22
                  C       GET POINTER TO FUNCTION BEAD STRUNG FROM LIST BEAD            REFORM     23
                    5   CALL DMGET(IWHOLE3,INPTR,NPTR)                                REFORM     24
                        IF (NPTR .EQ. 0) GO TO 9991                                     REFORM     25
     25           C                                                                     REFORM     26
                  C       CHECK IF EQUAL TYPE FUNCTION                                REFORM     27
                   10   CALL DMGET(IWHOLE2,NPTR,IFNCD)                                REFORM     28
                        IF (IFNCD .EQ. 5) GO TO 30                                      REFORM     29
                  C                                                                     REFORM     30
     30           C       GET NEW FUNCTION BEAD POINTER                                REFORM     31
                   20   CALL DMGET(IWHOLE1,NPTR,NPTR)                                 REFORM     32
                        IF (NPTR .EQ. 0) GO TO 70                                       REFORM     33
                        GO TO 10                                                        REFORM     34
                  C                                                                     REFORM     35
     35           C       GET NO OF ARGS IN EQ FUNCTION                                REFORM     36
                   30   CALL DMGET(IWHOLE3,NPTR,NARGS)                                REFORM     37
                  C                                                                     REFORM     38
                  C       GET PARAMETER NUMBER                                        REFORM     39
                        CALL DMGET(IWHOLE2,INPTR,NUMBP)                                 REFORM     40
     40           C                                                                     REFORM     41
                  C     *** DO FOR EACH ARGUEMENT                                     REFORM     42
                        IF (NARGS .EQ. 0) GO TO 70                                      REFORM     43
                        DO 60 J = 1,NARGS                                               REFORM     44
                  C                                                                     REFORM     45
     45           C       GET JTH ARG                                                  REFORM     46
                        CALL DMGET(IWHOLE4+J,NPTR,ARG)                                  REFORM     47
                        CALL PUTSYMB(IPTR,NUMBP,ARG,MAX)                                REFORM     48
                   60   CONTINUE                                                       REFORM     49
                        GO TO 20                                                        REFORM     50
     50           C                                                                     REFORM     51
                  C       UPDATE PARAMETER LIST POINTER                               REFORM     52
                   70   CALL DMGET(IWHOLE1,INPTR,INPTR)                               REFORM     53
                        IF (IPPTR .EQ. 0) GO TO 75                                      REFORM     54
                        GO TO 5                                                         REFORM     55
     55           C                                                                     REFORM     56
                  C       SEQUENTIAL ASSIGNMENT                                       REFORM     57
                   75   NPTR = IPTR                                                   REFORM     58
```

```
      C                                                                 REFORM     59
      C       PLACE SYMBOL IN PLOT IF NONE IS PRESENT                   REFORM     60
60       80 CALL DMGET(IWHOLE6,NPTR,ISYM)                               MODA      243
           IF (ISYM .NE. 0) GO TO 90                                     REFORM     62
           CALL DMSET(IWHOLE6,NPTR,MAX)                                  MODA      244
      C                                                                 REFORM     64
      C     UPDATE SYMBOL COUNT                                         REFORM     65
65         MAX = MAX + NOFYAX                                           REFORM     66
           IF (MAX+NOFYAX-1 .GT. 13) MAX =  1                           REFORM     67
      C                                                                 REFORM     68
      C     UPDATE POINTER                                              REFORM     69
        90 CALL DMGET(IWHOLE1,NPTR,NPTR)                                REFORM     70
70         IF (NPTR .EQ. 0) RETURN                                      REFORM     71
           GO TO 80                                                     REFORM     72
      C                                                                 REFORM     73
      C       ERROR CALLS                                               REFORM     74
       9991 CALL ERRDSM(40H(43H FN 80 MISSING (INTERNAL ERROR)-SEE ANALYST),  REFORM  75
75        X -3,6HASSIGN)                                                REFORM     76
           END                                                          REFORM     77
```

```
      SUBROUTINE AXSINIT, RETURNS(NEXT)                                 REFORM    78
C                                                                       REFORM    79
C        THIS SUBROUTINE UPDATES THE AXES BEAD POINTER SO THAT IT       REFORM    80
C        CONTAINS THE ADDRESS OF NEXT AXES BEAD TO PROCESS FOR A        REFORM    81
C        PARTICULAR PLOT GROUP. IF ONE IS NOT PRESENT A RETURN NEXT IS  REFORM    82
C        TAKEN SO THAT A NEW PG BEAD CAN BE FETCHED.                    REFORM    83
C                                                                       REFORM    84
      COMMON /ITABLE/ IELETAB,IFUNTAB,XSCL,YSCL                         MODA     245
      COMMON/CVECTOR/IPGPT,ICONPT,IAXSPT,IPMPT                          REFORM    85
      COMMON/MNINFO/ NOFYAX,LENGTH,MSTRUCT,NTABL,INOXPTR,NOINOX,LAST    REFORM    86
      COMMON/STATSWT/ ISTAT,ISTATM,ISTATP,STRSWT                        REFORM    87
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                REFORM    88
      COMMON/COMCODS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   REFORM    89
     X IWHOLE6                                                          REFORM    90
      COMMON /TRIM/ NAXES,NSTOR(3)                                      TRIM      2
C                                                                       AB        2
C        AXIS BEAD                                                      AB        3
      COMMON /AB/ IABNXT,IABLST,IABT,IABPM,IABXN,IABNY                  AB        4
C                                                                       REFORM    91
C        CHECK WETHER INITIALIZATION OR UPDATE                          REFORM    92
      IF (ISTAT) GO TO 30                                               REFORM    93
C                                                                       REFORM    94
C        UPDATE CURRENT AXES POINTER                                    REFORM    95
      LAXSPT= IAXSPT                                                    MODB      1
      CALL DMGET(IWHOLE1,IAXSPT,IAXSPT)                                 REFORM    96
C                                                                       REFORM    97
C        NEXT BEAD PRESENT                                              REFORM
      IF (IAXSPT .NE. 0) GO TO 20                                       MODA     24
      IF (NAXES .EQ. 3) CALL TRIM                                       MODA     249
      RETURN NEXT                                                       MODA     250
   20 CONTINUE                                                          MODA     251
      ISTAT = .TRUE.                                                    REFORM   100
      GO TO 40                                                          REFORM   101
C                                                                       REFORM   102
C        INITIALIZE AXES POINTER FROM PG BEAD                           REFORM   103
   30 CALL DMGET(IWHOLE4,IPGPT,IAXSPT)                                  REFORM   104
      NAXES = 0                                                         MODA     252
C                                                                       REFORM   105
C     *** INITIALIZE AXES INFORMATION                                   REFORM   106
C                                                                       REFORM   107
C        GET NUMBER OF Y AXES                                           REFORM   108
   40 CALL DMGET(IWHOLE3,IAXSPT,NOFYAX)                                 REFORM   109
      NAXES = NAXES + 1                                                 MODA     253
      IF (NAXES .LE. 3) NSTOR(NAXES) = IAXSPT                           MODA     254
      CALL SCLBD(IAXSPT,NOFYAX,LAST)                                    MODA     255
C                                                                       MODA     256
C        RESTRUCTURE AXIS BEAD                                          MODA     257
      CALL DMGET(IWHOLE5+NOFYAX,IAXSPT,IATTMP)                          MODA     258
      DO 50 I=1,NOFYAX                                                  MODA     259
      J = NOFYAX - I + 1                                                MODA     260
      CALL DMGET(IWHOLE4+J,IAXSPT,ITMP)                                 MODA     261
      CALL SEARCH(IELETAB,2,ITMP)                                       MODA     262
      IF (ITMP .EQ. 0) GO TO 9991                                       MODA     263
      ITMP = LOADVAL(IELETAB,1,ITMP)                                    MODA     264
      CALL DMSET(IABNY+J ,IAXSPT,ITMP)                                  MODA     265
   50 CONTINUE                                                          MODA     266
      CALL DMGET(IWHOLE3,IAXSPT,ITMP)                                   MODA     267
```

```
      CALL DMSET(IA8NY,IAXSPT,ITMP)                                    MODA   268
      CALL DMGET(IWHOLE2,IAXSPT,ITMP)                                  MODA   269
60    CALL SEARCH(IELETA8,2,ITMP)                                      MODA   270
      IF (ITMP .EQ. 0) GO TO 9991                                      MODA   271
      ITMP = LOADVAL(IELETA8,1,ITMP)                                   MODA   272
      CALL DMSET(IA8XN,IAXSPT,ITMP)                                    MODA   273
      CALL DMSET(IA8LST,IAXSPT,LAXSPT)                                 MODB     2
65    CALL DMSET(IA8T,IAXSPT,IATTMP)                                   MODA   275
      RETURN                                                           REFORM 110
9991  CALL ERROR*(31H(26H ELE NO NOT FOUND IN TABLE),-3,7HAXSINIT)     MODA   276
      END                                                              REFORM 111
```

```
      1                SUBROUTINE CHNGC8(IPTR)                                          MODA      277
                       COMMON /ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                      MODA      278
                       COMMON /CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5, MODA      279
                      X IWHOLE6                                                         MODA      280
      5          C                                                                     C8        2
                 C        CURVE BEAD                                                   C8        3
                       COMMON /C8/ IC8NXT,IC8LST,IC8PN,IC8PV,IC8NPT,IC8SYM,IC8R8,IC8X8, C8        4
                      X            IC8Y8                                              C8        5
                       COMMON /PTADSZ/ NOFPTS                                           MODA      282
     10                DIMENSION XPTS(250),YPTS(250),RECNO(250)                         MODA      283
                       DIMENSION TABLE(250,3)                                           MODA      284
                       EQUIVALENCE (XPTS(1),TABLE(1,1)), (YPTS(1),TABLE(1,2)),          MODA      285
                      X (RECNO(1),TABLE(1,3))                                           MODA      286
                 C                                                                     MODA      287
     15          C        RESTRUCTURE CURVE BEAD                                       MODA      288
                       CALL DMGET(IWHOLE0+6,IPTR,ITMPY)                                 MODA      289
                       CALL DMSET(IC8Y8,IPTR,ITMPY)                                     MODA      290
                       CALL DMGET(IWHOLE0+7,IPTR,ITMPX)                                 MODA      291
                       CALL DMSET(IC8X8,IPTR,ITMPX)                                     MODA      292
     20                CALL DMGET(IWHOLE6,IPTR,ITMPR)                                   MODA      293
                       CALL DMSET(IC8R8,IPTR,ITMPR)                                     MODA      294
                       CALL DMGET(IWHOLE4,IPTR,ITMP)                                    MODA      295
                       CALL DMSET(IC8SYM,IPTR,ITMP)                                     MODA      296
                       CALL DMGET(IWHOLE2,IPTR,ITMP)                                    MODA      297
     25                CALL DMSET(IWHOLE2,IPTR,0)                                       MODA      298
                       CALL DMSET(IC8PV,IPTR,ITMP)                                      MODA      299
                       CALL DMGET(IWHOLE5,IPTR,ITMP)                                    MODA      300
                       CALL DMGET(IWHOLE3,IPTR,ITEMP)                                   MODA      301
                       CALL DMSET(IC8NPT,IPTR,ITEMP)                                    MODA      302
     30                CALL SEARCH(IELETAB,2,ITMP)                                      MODA      303
                       IF (ITMP .EQ. 0) GO TO 9991                                      MODA      304
                       ITMP = LOADVAL(IELETAB,1,ITMP)                                   MODA      305
                       CALL DMSET(IC8PN,IPTR,ITMP)                                      MODA      306
                       WRITE(6,10) IPTR                                                 MODA      307
     35             10 FORMAT(1X,*AT CB *,O10)                                          MODA      308
                 C                                                                     MODA      309
                 C        RETRIEVE X,Y COORDS AND SORT ON X VALUES                     MODA      310
                       MAXPTS = 248                                                     MODA      311
                       NPTS = ITEMP                                                     MODA      312
     40             20 IF (NPTS .LE. 0) RETURN                                          MODA      313
                       NUSE = MAXPTS                                                    MODA      314
                       IF (NPTS .LT. MAXPTS) NUSE = NPTS                                MODA      315
                       ICOMP = 0700000000013 + SHIFT(NUSE,18)                           MODA      316
                       CALL DMGET(ICOMP,ITMPX,XPTS)                                     MODA      317
     45                CALL DMGET(ICOMP,ITMPY,YPTS)                                     MODA      318
                       CALL DMGET(ICOMP,ITMPR,RECNO)                                    MODA      319
                       CALL SSORT(TABLE,250,3,NUSE,1)                                   MODA      320
                       CALL DMSET(ICOMP,ITMPX,XPTS)                                     MODA      321
                       CALL DMSET(ICOMP,ITMPY,YPTS)                                     MODA      322
     50                CALL DMSET(ICOMP,ITMPR,RECNO)                                    MODA      323
                       CALL DMGET(IWHOLE1+NOFPTS,ITMPX,ITMPX)                           MODA      324
                       CALL DMGET(IWHOLE1+NOFPTS,ITMPY,ITMPY)                           MODA      325
                       CALL DMGET(IWHOLE1+NOFPTS,ITMPR,ITMPR)                           MODA      326
                       NPTS = NPTS - NUSE                                               MODA      327
     55                GO TO 20                                                         MODA      328
                 C                                                                     MODA      329
                 C        ERROR CALLS                                                  MODA      330
```

```
9991 CALL ERRORM(31H(26H ELE NO NOT FOUND IN TABLE),-3,6HCHNGCB)      MODA    331
     END                                                              MODA    332
```

```
      SUBROUTINE FILPLT(IPLOT,IC8PTR)                                   MODA   333
C                                                                       REFORM 113
C        THIS SUBROUTINE BUILDS AND FILLS A PLOT BEAD FOR A PARTICULAR  REFORM 114
C        SET OF CURVE BEADS                                             REFORM 115
C                                                                       REFORM 116
C        IC8PTR - THE POINTER TO THE FIRST CURVE BEAD                   REFORM 117
C        IPLOT - POINTER TO FIRST PLOT BEAD                             REFORM 120
C                                                                       REFORM 121
C              *** PLOT BEAD FORMAT ***                                 REFORM 122
C        WORD1 * PTR TO NEXT          *                                 REFORM 123
C        WORD2 * TAPE NO.             *                                 REFORM 124
C        WORD3 * PLOT GROUP POINTER   *                                 REFORM 125
C        WORD4 * TITLE TEXT POINTER    *                                REFORM 126
C        WORD5 * AXIS BEAD POINTER     *                                REFORM 127
C        WORD6 * AXIS SCALE POINTER    *                                REFORM 128
C        WORD7 * PARA-MULT POINTER     *                                REFORM 129
C        WORD8 * TAPE BLOCK NUMBER     *                                REFORM 130
C        WORD9 * CURVE BEAD POINTER    *                                REFORM 131
C        WORD10* LABEL LIST POINTER   *                                 REFORM 132
C        WORD11* MULTIPLIER 1 POINTER*                                  REFORM 133
C        WORDXX*       ...            *                                REFORM 134
C        WORDNN* MULTIPLIER N POINTER*                                  REFORM 135
C              * * * * * * * * * * * **                                 REFORM 136
      COMMON/MDINFO/ NOFYAX,LENGTH,MSTRUCT,NTPBL,INDXPTR,NOINDX,LSTPTR  REFORM 137
      COMMON/CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   REFORM 138
     X IWHOLE6                                                          REFORM 139
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT                         REFORM 140
C                                                                       REFORM 154
C        FILL PLOT BEAD                                                 REFORM 155
    5 CONTINUE                                                          REFORM 156
      IP8PTR = IPLOT                                                    MODA   334
      CALL DMSET(IWHOLE5,IP8PTR,IPGPT)                                  MODA   335
      CALL DMGET(IWHOLE3,IPGPT,IPTR)                                    MODA   336
      CALL DMSET(IWHOLE6,IP8PTR,IPTR)                                   MODA   337
      CALL DMSET(IWHOLE0+7,IP8PTR,IAXSPT)                               MODA   338
      CALL DMSET(IWHOLE0+9,IP8PTR,IPMPT)                                MODA   339
      CALL DMSET(IWHOLE0+11,IP8PTR,IC8PTR)                              MODA   340
C                                                                       REFORM 164
C        CREATE SCALE BEAD AND PLACE PTR IN PLOT BEAD                   REFORM 165
      ISCLBD = LSTPTR                                                   MODA   341
      CALL DMSET(IWHOLE0+8,IP8PTR,ISCLBD)                               MODA   342
C                                                                       REFORM 168
C        FILL MULTIPLIER SECTION OF BEAD                                REFORM 169
      IF (LENGTH .EQ. 0) GO TO 20                                       REFORM 170
      CALL DMSET(IWHOLE0+12,IP8PTR,LENGTH)                              MODA   343
      ILEVCNT = 1                                                       REFORM 171
      CALL DMGET(IWHOLE4,IPMPT,IPTR)                                    MODA   344
   10 CALL DMGET(IWHOLE2,IPTR,ITMP)                                     MODA   345
      CALL DMSET(IWHOLE0+11+2*ILEVCNT,IP8PTR,ITMP)                      MODA   346
      IF (ILEVCNT.EQ.LENGTH) GO TO 20                                   REFORM 174
      ILEVCNT = ILEVCNT + 1                                             REFORM 175
      CALL DMGET(IWHOLE1,IPTR,IPTR)                                     MODA   347
      GO TO 10                                                          REFORM 176
   20 CONTINUE                                                          MODA   348
      RETURN                                                            MODA   349
      END                                                               REFORM 221
```

```
1                REAL FUNCTION GETVAL(ITABPTR,ITABCOL,ITABROW)                    REFORM   222
                 EQUIVALENCE (N,GETV)                                             REFORM   223
                 N = LOADVAL(ITABPTR,ITABCOL,ITABROW)                             REFORM   224
                 GETVAL=GETV                                                      REFORM   225
5                RETURN                                                           REFORM   226
                 END                                                              REFORM   227
```

```
      INTEGER FUNCTION LOADVAL(ITABPTR,ITABCOL,ITABROW)                  REFORM    226
      COMMON /CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWH5,IWH6  REFORM    229
C                                                                        REFORM    230
C        GET POINTER TO CORRECT TABLE COL                                REFORM    231
      CALL DMGET(IWHOLE2+ITABCOL,ITABPTR,ICOLPTR)                        REFORM    232
C                                                                        REFORM    233
C        GET VALUE FROM SELECTED ROW                                     REFORM    234
      CALL DMGET(IWHOLE0+ITABROW,ICOLPTR,LOADVAL)                        REFORM    235
      RETURN                                                             REFORM    236
      END                                                                REFORM    237
```

```
      SUBROUTINE MLTINIT, RETURNS(NEXT)                                 REFORM   238
C                                                                       REFORM   239
C     THIS SUBROUTINE UPDATES THE MULTIPLIER STRUCTURE BEAD FROM THE    REFORM   240
C     LOWEST LEVEL UPWARD TO CONTAIN THE ADDRESS REPRESENTATION OF      REFORM   241
C     SEQUENTIAL MOVEMENT TO THE NEXT PARAMETER VALUE BEAD. WHEN THIS   REFORM   242
C     REACHES ITS LIMIT A RETURN NEXT IS TAKEN.                         REFORM   243
C                                                                       REFORM   244
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                         REFORM   245
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                REFORM   246
      COMMON/CVECTOR / IPGPT, ICONPT, IAXSPT, IPMPT                     REFORM   247
      COMMON/MNINFO/ NOFYAX,LENGTH,MSTRUCT,NTPBL,INDXPTR,NOINDX,LAST    REFORM   248
      COMMON/CMCODS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   REFORM   249
     X IWHOLE6                                                          REFORM   250
C                                                                       REFORM   252
C        UPDATE OR INITIALIZATION                                       REFORM   253
      IF (ISTAT) GO TO 30                                               REFORM   254
C                                                                       REFORM   255
C        CHECK FOR MULTIPLIERS                                          REFORM   256
      IF (.NOT. STRSWT) RETURN NEXT                                     MODA     350
C                                                                       REFORM   258
C        UPDATE PLOT BEAD POINTER (MSTRUCT)                             MODA     351
      LSTRUCT = MSTRUCT                                                 MODB       3
      CALL DMGET(IWHOLE1,MSTRUCT,MSTRUCT)                               MODA     352
      IF (MSTRUCT .EQ. 0) RETURN NEXT                                   MODA     353
      CALL DMSET(IWHOLE3,MSTRUCT,LSTRUCT)                               MODB       4
      ISTAT = .TRUE.                                                    MODA     354
      RETURN                                                            MODA     355
C                                                                       MODA     356
C        INITIALIZE MSTRUCT IF PLOT BEADS PRESENT                       MODA     357
   30 MSTRUCT = 0                                                       MODA     358
      IF (.NOT. STRSWT) RETURN NEXT                                     MODA     359
      CALL DMGET(IWHOLE5,IPMPT,MSTRUCT)                                 MODA     360
      RETURN                                                            MODA     361
      END                                                               REFORM   299
```

```
      SUBROUTINE PGINIT,RETURNS(FIN)                                    REFORM    300
C                                                                       REFORM    301
C        THIS SUBROUTINE UPDATES THE PLOT GROUP POINTER TO CONTAIN THE   REFORM    302
C        ADDRESS OF THE NEXT PG BEAD IN THE DATA STRUCTURE. IF NONE IS  REFORM    303
C        FOUND A RETURN IS MADE VIA FIN. OTHERWISE THE ROUTINE WILL     REFORM    304
C        CONTINUE AND DELETE THE CURRENT COND LIST BEFORE RETURNING.    REFORM    305
C                                                                       REFORM    306
      COMMON/HEADPTR/IFPG                                               REFORM    307
      COMMON/CVECTOR/IPGPT,ICONPT,IAXSPT,IPMPT                          REFORM    308
      COMMON/STATSWT/ ISTAT,ISTATM,ISTATP,STRSWT                        REFORM    309
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                REFORM    310
      COMMON/CAPCDS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,    REFORM    311
     X IWHOLE6                                                          REFORM    312
C                                                                       PG          2
C        PLOT GROUP BEAD                                                PG          3
      COMMON /PG/ IPGNXT,IPGLST,IPGTT,IPGAB,IPGCT,IPGNCT,IPGLAB,        PG          4
     X            IPGNLB,IPGLB                                         PG          5
C                                                                       REFORM    313
C        CHECK IF FIRST INITIALIZATION                                  REFORM    314
       IF (.NOT. ISTAT) GO TO 10                                        REFORM    315
C                                                                       REFORM    316
C        INITIALIZE PG POINTER                                          REFORM    317
      IPGPT = IFPG                                                      REFORM    318
      GO TO 20                                                          REFORM    319
C                                                                       REFORM    320
C        NOT FIRST INITIALIZATION                                       REFORM    321
   10 ISTAT = .TRUE.                                                    REFORM    322
C                                                                       REFORM    323
C        ZERO OUT CVECTOR                                               REFORM    324
      ICONPT=IAXSPT= IPMPT = 0                                          REFORM    325
C                                                                       REFORM    326
C        GET NEXT PG BEAD                                               REFORM    327
      LPGPT = IPGPT                                                     MODB        5
      CALL DMGET(IWHOLE1,IPGPT,IPGPT)                                   REFORM    328
C                                                                       REFORM    329
C        END OF DATA STRUCTURE, RETURN FIN                              REFORM    330
      IF (IPGPT .EQ. 0) RETURN FIN                                      REFORM    331
C                                                                       REFORM    332
C        UPDATE POINTER TO CONDITION TEXT BEAD                          REFORM    333
   20 CALL DMGET(IWHOLE5,IPGPT,ICONPT)                                  REFORM    334
C                                                                       REFORM    335
C        DELETE CONDITION LIST & ASSOCIATED FN BEADS                    REFORM    336
      CALL DMGET(IWHOLE3,IPGPT ,LSTPTR)                                 REFORM    337
      IF (LSTPTR .EQ. 0) GO TO 30                                       REFORM    338
      CALL FRNLIST(LSTPTR,0)                                            REFORM    339
C                                                                       REFORM    340
C        ZERO OUT THIRD WORD OF PG BEAD                                 REFORM    341
      CALL DMSET(IWHOLE3,IPGPT,0)                                       REFORM    342
C                                                                       REFORM    343
C        DELETE LABEL REQUEST BEADS                                     REFORM    344
   30 CALL DMGET(IWHOLE0+7,IPGPT,LSTPTR)                                REFORM    345
      IF (LSTPTR .EQ. 0) GO TO 40                                       REFORM    346
      CALL FRNLIST(LSTPTR,0)                                            REFORM    347
      CALL DMSET(IWHOLE0+7,IPGPT,0)                                     REFORM    348
   40 CONTINUE                                                          MODA      363
C                                                                       MODA      364
C        RESTRUCTURE PLOT GROUP BEAD                                    MODA      365
```

```
      CALL CMGET(IWHOLE2,IPGPT,ITEMP)                    MODA    366
      CALL CMSET(IPGTT,IPGPT,ITEMP)                      MODA    367
60    CALL CMSET(IPGLST,IPGPT,LPGPT)                     MODB      6
      CALL CMGET(IWHOLEC+7,IPGPT,ITEMP)                  MODA    369
      CALL CMSET(IPGLB,IPGPT,ITEMP)                      MODA    370
      CALL CMSET(IPGLAB,IPGPT,0)                         MODA    371
      RETURN                                             MODA    372
65    END                                                REFORM  3>0
```

```
      SUBROUTINE PMINIT,RETURNS(NEXT)                                   REFORM   351
C                                                                       REFORM   352
C     THIS SUBROUTINE UPDATES THE PM POINTER TO CONTAIN THE ADDRESS OF  REFORM   353
C        THE NEXT PM FOR A GIVEN AXES BEAD. IF NONE IS FOUND A RETURN   REFORM   354
C     NEXT IS TAKEN.                                                    REFORM   355
C                                                                       REFORM   356
      COMMON /ITABLE/ IELETAB,IFUNTAB,XSCL,YSCL                         MODA     373
      COMMON/STATSWT/ ISTAT,ISTATM,ISTATP,STRSWT                        REFORM   357
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                REFORM   358
      COMMON /CVECTOR/ IPGPT,ICONPT,IAXSPT,IPM                          MODA     374
      COMMON/ CMPCOS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   REFORM   360
     X IWHOLE6                                                          REFORM   361
C                                                                       PM         2
C        PARAMETER - MULTIPLIER BEAD                                    PM         3
      COMMON /PM/ IPMNXT,IPMLST,IPMPL,IPMML,IPMPB,IPMPT,IPMMT           PM         4
C                                                                       REFORM   362
C        INITIALIZATION OR UPDATE                                       REFORM   363
      IF (ISTAT) GO TO 30                                               REFORM   364
C                                                                       REFORM   365
C        UPDATE CURRENT PM POINTER                                      REFORM   366
      LPM = IPM                                                         MODB       7
      CALL DMGET(IWHOLE1,IPM,IPM)                                       MODA     376
C                                                                       REFORM   368
C        CHECK IF PM BEADS ARE EXHAUSTED FOR CURRENT AXES BEAD          REFORM   369
      IF (IPM .EQ. 0) RETURN NEXT                                       MODA     377
      ISTAT = .TRUE.                                                    REFORM   371
      GO TO 50                                                          REFORM   372
C                                                                       REFORM   373
C        INITIALIZE PM BEAD FROM NEW AXES BEAD                          REFORM   374
   30 CALL DMGET(IWHOLE4,IAXSPT,IPM)                                    MODA     378
C                                                                       REFORM   376
C        RESET LOGICAL SWITCHES AND MSTRUCT BEAD                        REFORM   377
   50 CALL RESET                                                        REFORM   378
C                                                                       MODA     379
C        RESTRUCTURE P-M BEAD                                           MODA     380
      CALL DMGET(IWHOLE3,IPM  ,ITMP)                                    MODA     381
      CALL DMSET(IPMML,IPM  ,ITMP)                                      MODA     382
   60 IF (ITMP .EQ. 0) GO TO 65                                         MODA     383
      CALL DMGET(IWHOLE2,ITMP,NAME)                                     MODA     384
      CALL SEARCH(IELETAB,2,NAME)                                       MODA     385
      IF (NAME .EQ. 0) GO TO 9991                                       MODA     386
      NAME = LOADVAL(IELETAB,1,NAME)                                    MODA     387
      CALL DMSET(IWHOLE2,ITMP,NAME)                                     MODA     388
      CALL DMGET(IWHOLE1,ITMP,ITMP)                                     MODA     389
      GO TO 60                                                          MODA     390
   65 CONTINUE                                                          MODA     391
      CALL DMGET(IWHOLE2,IPM  ,ITMP)                                    MODA     392
      CALL DMSET(IPMPL,IPM  ,ITMP)                                      MODA     393
      CALL DMSET(IPMLST,IPM,LPM)                                        MODB       8
   70 IF (ITMP .EQ. 0) GO TO 75                                         MODA     395
      CALL DMGET(IWHOLE2,ITMP,NAME)                                     MODA     396
      CALL SEARCH(IELETAB,2,NAME)                                       MODA     397
      IF (NAME .EQ. 0) GO TO 9991                                       MODA     398
      NAME = LOADVAL(IELETAB,1,NAME)                                    MODA     399
      CALL DMSET(IWHOLE2,ITMP,NAME)                                     MODA     400
      CALL DMGET(IWHOLE1,ITMP,ITMP)                                     MODA     401
      GO TO 70                                                          MODA     402
```

```
          75 CONTINUE                                                          MODA    403
             RETURN                                                            REFORM  379
60        9991 CALL ERRORM(31H(26H ELE NO NOT FOUND IN TABLE),-3,6HPMINIT)      MODA    404
             END                                                               REFORM  380
```

```
      SUBROUTINE PRNLIST(LSTPTR,IND)                                    REFORM  381
C                                                                       REFORM  382
C        IND=0 - PRUNE LIST AND FN BEADS                                REFORM  383
C        IND=1 - PRUNE FN BEADS ONLY                                    REFORM  384
C        LSTPTR - POINTER TO BEGINNING OF CONNECTED 3 WORD LIST BEADS   REFORM  385
C                                                                       REFORM  386
      COMMON/CMFCDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   REFORM  387
     X IWHOLE6                                                          REFORM  388
C                                                                       REFORM  389
C        GET FN POINTER                                                 REFORM  390
   10 CALL DMGET(IWHOLE3,LSTPTR,IFNPTR)                                 REFORM  391
      CALL DMSET(IWHOLE3,LSTPTR,0)                                      REFORM  392
C                                                                       REFORM  393
C        LOOP THROUGH FN LIST DELETING ALL BEADS                        REFORM  394
   20 IF (IFNPTR .EQ. 0) GO TO 30                                       REFORM  395
      CALL DMGET(IWHOLE1,IFNPTR,NFNPTR)                                 REFORM  396
      CALL DMRLBD(IFNPTR)                                               REFORM  397
      IFNPTR = NFNPTR                                                   REFORM  398
      GO TO 20                                                          REFORM  399
C                                                                       REFORM  400
C        LOOP TO NEXT LIST BEAD DELETING CURRENT LIST BEAD IF REQUESTED REFORM  401
   30 CALL DMGET(IWHOLE1,LSTPTR,NLSTPTR)                                REFORM  402
      IF (IND .EQ. 0) CALL DMRLBD(LSTPTR)                               REFORM  403
      LSTPTR = NLSTPTR                                                  REFORM  404
      IF (LSTPTR .NE. 0) GO TO 10                                       REFORM  405
      RETURN                                                            REFORM  406
      END                                                               REFORM  4
```

```
      PROGRAM PSPMAIN                                                    REFORM   408
C                                                                        REFORM   409
C        THIS PROGRAM IS THE DRIVER ROUTINE FOR THE INITIATION AND       REFORM   410
C        UPDATE OF POINTERS TO THE MAIN SECTIONS OF THE LIST STRUCTURE   REFORM   411
C        AND ALSO CONTROLS THE PRUNING OF UNNECESSARY LIST BEADS AND THE REFORM   412
C        RESTRUCTURING OF THE CURRENT CURVE BEADS.                       REFORM   413
C                                                                        REFORM   414
      COMMON /HEADPTR/IFPG                                               REFORM   415
      COMMON /ITABLE/ IELETAB,IFUNTAB,XSCL,YSCL                          REFORM   416
      COMMON/CVECTOR/IPGPT,ICONPT,IAXSPT,IPMPT                           REFORM   417
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                          REFORM   418
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                 REFORM   419
      COMMON/MNINFO/NOFYAX,LENGTH,MSTRUCT,NTPBL,INDXPTR,NOINDX,LAST      REFORM   420
      COMMON /CMPCDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   REFORM   421
     X IWHOLE6                                                           REFORM   422
      DATA IPGPT,ICONPT,IAXSPT,IPMPT/4*(0)/                              REFORM   423
      DATA ISTAT,ISTATM,ISTATP/.TRUE.,.FALSE.,.FALSE./                   REFORM   424
      DATA LENGTH,MSTRUCT,NTPBL,INDXPTR,NOINDX/0,0,3,0,0/                REFORM   425
      DATA IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6       REFORM   426
     X/0100000000B,0100000000018,0100000000028,0100000000038,            REFORM   427
     X 0100000000048,0100000000058,0100000000068/                        REFORM   428
C                                                                        REFORM   429
C        SORT ELEMENT TABLE ON ELEMENT NUMBER                            REFORM   430
      CALL SORTAB(IELETAB,2)                                             REFORM   431
C                                                                        REFORM   432
C        UPDATE PLOT GROUP POINTER                                       REFORM   433
   10 CALL PGINIT,RETURNS(50)                                            REFORM   434
      WRITE(6,11)                                                        MODA     405
   11 FORMAT(* AFTER PGINIT*)                                            MODA     406
C                                                                        REFORM   435
C        UPDATE REMAINING DATA STRUCTURE POINTERS, PRUNING THE STRUCTURE REFORM   436
C        WHENEVER POSSIBLE                                               REFORM   437
   20 CALL AXSINIT,RETURNS(10)                                           REFORM   438
      WRITE(6,21)                                                        MODA     407
   21 FORMAT(* AFTER AXSINIT*)                                           MODA     408
   30 CALL PMINIT,RETURNS(20)                                            REFORM   439
      WRITE(6,31)                                                        MODA     409
   31 FORMAT(* AFTER PMINIT*)                                            MODA     410
   40 CALL MLTINIT,RETURNS(30)                                           REFORM   440
      WRITE(6,41)                                                        MODA     411
   41 FORMAT(* AFTER MLTINIT*)                                           MODA     412
C                                                                        REFORM   441
C        REGROUP CURVE BEADS UNDER NEW PLOT BEAD AND PERFORM SYMBOL      REFORM   442
C        ASSIGNMENT                                                      REFORM   443
      CALL RESTRCT                                                       REFORM   444
      WRITE(6,51)                                                        MODA     413
   51 FORMAT(* AFTER RESTRCT*)                                           MODA     414
      GO TO 40                                                           REFORM   445
C                                                                        REFORM   446
C        BUILD GRAPHICS INDEX AND SET UP POINTERS FOR NEXT OVERLAY       REFORM   447
   50 CONTINUE                                                           MODA     415
      CALL DMSET(IWHOLE2,100003,IELETAB)                                 MODA     416
      CALL DMSET(IWHOLE1,100009,IFPG)                                    MODA     417
      END                                                                REFORM   449
```

```
      SUBROUTINE PUTSYM3(NPTR,NUM3P,ARG,MAX)                                REFORM   450
C                                                                           REFORM   451
C       THIS SUBROUTINE SEARCHS THE PLOT BEAD LIST INDICATED BY NPTR        REFORM   452
C       LOOKING FOR A MATCH IN THE SECOND WORD OF THE PLOT BEAD WITH        REFORM   453
C       ARG. WHEN A MATCH IS FOUND A SYMBOL DENOTED BY MAX IS ASSIGNED.     REFORM   454
C                                                                           REFORM   455
      COMMON/CMECDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,       REFORM   456
     X IWHOLE6                                                              REFORM   457
      IPTR = NPTR                                                           REFORM   458
C                                                                           REFORM   459
C       GET PARAMETER NUMBER FROM PLOT BEAD                                 REFORM   460
   10 CALL DMGET(IWHOLE3,IPTR,INUM)                                         MODA     418
C                                                                           REFORM   462
C       COMPARE WITH NUM3P                                                  REFORM   463
      IF (INUM .NE. NUM3P) GO TO 30                                         REFORM   464
C                                                                           REFORM   465
C       GET PARAMETER VALUE FROM BEAD                                       REFORM   466
      CALL DMGET(IWHOLE4,IPTR,PVAL)                                         MODA     419
C                                                                           REFORM   468
C       COMPARE WITH ARG                                                    REFORM   469
      IF (PVAL .NE. ARG) GO TO 30                                           REFORM   470
C           EQUAL                                                           REFORM   471
C       CHECK FOR PREVIOUS SYMBOL                                           REFORM   472
      CALL DMGET(IWHOLE5,IPTR,ISYM)                                         MODA     420
      IF (ISYM .EQ. 0) CALL DMSET(IWHOLE6,IPTR,MAX)                         MODA     421
C                                                                           REFORM   475
C       UPDATE SYMBOL                                                       REFORM   476
   20 MAX = MAX + 1                                                         REFORM   477
      IF (MAX .GT. 13)  MAX = 1                                             REFORM   478
      RETURN                                                                REFORM   479
C                                                                           REFORM   480
C       UPDATE POINTER TO PLOT BEAD                                         REFORM   481
   30 CALL DMGET(IWHOLE1,IPTR,IPTR)                                         REFORM   482
      IF (IPTR .EQ. 0) GO TO 20                                             REFORM   483
      GO TO 10                                                              REFORM   484
      END                                                                   REFORM   485
```

```
      SUBROUTINE RESET                                                  REFORM   486
C                                                                       REFORM   487
C        THIS SUBROUTINE INITIALIZES LENGTH; LOGICAL SWITCHES           REFORM   488
C        ISTATM, ISTATP, STRSWT; AND FREES MSTRUCT IF PREVIOUSLY USED.  REFORM   489
C                                                                       REFORM   490
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                REFORM   491
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                         REFORM   492
      COMMON/CVECTOR/IPGPT,ICONPT,IAXSPT,IPMPT                          REFORM   493
      COMMON/CHECOS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,    REFORM   494
     X IWHOLE6                                                          REFORM   495
      COMMON/MNINFO/NOFVAX,LENGTH,MSTRUCT,NTPBL,INDXPTR,NOINDX,LAST     REFORM   496
C                                                                       REFORM   506
C        SET STRUCTURE SWITCH                                           REFORM   507
   10 STRSWT = .TRUE.                                                   REFORM   508
      CALL DMGET(IWHOLE5,IPMPT,IPTR)                                    REFORM   509
      IF (IPTR .EQ. 0) STRSWT = .FALSE.                                 REFORM   510
C                                                                       REFORM   511
C        GET LENGTH OF MULTIPLIER LIST FROM CURRENT PM BEAD             REFORM   512
      CALL DMGET(IWHOLE4,IPMPT,LENGTH)                                  REFORM   513
C                                                                       REFORM   514
C        SET MULT REQUEST SWITCH ACCORDINGLY                            REFORM   515
      ISTATM = .FALSE.                                                  MODA     422
      IF (LENGTH .NE. 0) ISTATM = .TRUE.                                REFORM   516
C                                                                       REFORM   517
C        CHECK IF ANY PARAMETERS EXIST FOR PM BEAD                      REFORM   518
      ISTATP = .FALSE.                                                  REFORM   519
      CALL DMGET(IWHOLE2,IPMPT,IPTR)                                    REFORM   5[illegible]
      IF(IPTR .EQ. 0) GO TO 30                                          REFORM   5[illegible]
C                                                                       REFORM   522
C                                                                       REFORM   523
C        SET ON PARAMETER INDICATOR SWITCH                              REFORM   524
      ISTATP = .TRUE.                                                   REFORM   525
   30 RETURN                                                            REFORM   526
      END                                                               REFORM   527
```

```
      SUBROUTINE RESTRCT                                                REFORM   528
C                                                                       REFORM   529
C        THIS SUBROUTINE CONTROLS THE SEGMENTING OF THE CURVE BEADS INTO REFORM   530
C        A PLOT BEAD LEVEL AND ALSO THE CURVE BEAD SYMBOL ASSIGNMENT.   REFORM   531
C                                                                       REFORM   532
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                         REFORM   533
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                REFORM   534
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT                         REFORM   535
      COMMON/MAINFO/ NOFYAX,LENGTH,MSTRUCT,NTPBL,INDXPTR,NOINDX,LAST    REFORM   536
      COMMON/CMPCDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   REFORM   537
     X IWHOLE6                                                          REFORM   538
C                                                                       REFORM   539
C        INITIALIZE VARIABLES                                           REFORM   540
      IPTR = MSTRUCT                                                    MODA     424
C                                                                       REFORM   550
C        CHECK WHETHER PLOT BEAD AND DATA POINTS EXIST                  REFORM   551
      IF (IPTR .EQ. 0) GO TO 70                                         MODA     425
C                                                                       REFORM   559
C        SAVE BEGINNING PTR TO PLOT BEAD                                REFORM   560
   20 IPLOT = IPTR                                                      REFORM   561
C                                                                       REFORM   562
C        STEP DOWN PLOT BEAD TO FIRST CURVE BEAD                        REFORM   563
      CALL DMGET(IWHOLE0+11,IPTR,IPTR)                                  MODA     426
C                                                                       MODA     427
C        FILL IN REMAINDER OF PLOT BEAD                                 MODA     428
      CALL FILPLT(IPLOT,IPTR)                                           MODA     429
C                                                                       REFORM   583
C        SAVE BEGINNING POINTER TO CURVE BEAD                           REFORM   584
      IOPTR = IPTR                                                      REFORM   585
C                                                                       REFORM   586
C        RESTRUCTURE CURVE BEADS AND SORT POINT VALUES                  MODA     430
      LPTR = 0                                                          MOD3       9
   30 CONTINUE                                                          MODA     431
      CALL CHNGCB(IPTR)                                                 MODA     432
      IF (LPTR .NE. 0) CALL DMSET(IWHOLE2,IPTR,LPTR)                    MOD8      10
C        GET NEXT CURVE BEAD                                            MODA     433
      LPTR = IPTR                                                       MOD8      11
      CALL DMGET(IWHOLE1,IPTR,IPTR)                                     REFORM   591
      IF (IPTR .NE. 0) GO TO 30                                         MODA     434
C                                                                       MODA     435
C        ASSIGN SYMBOLS TO CURVES OF PLOT BEAD                          MODA     436
      CALL ASSIGN(IOPTR)                                                MODA     437
C                                                                       REFORM   607
C        SET STATUS SWITCH TO INDICATE UPDATE NEEDED                    REFORM   608
   70 ISTAT = .FALSE.                                                   REFORM   609
      RETURN                                                            REFORM   610
      END                                                               REFORM   611
```

```
      1                 SUBROUTINE SCL80(IAXSPT,NOFYAX,ISCL80)                                REFORM    612
                  C                                                                         REFORM    613
                  C       THIS ROUTINE IS USED TO CREATE AND FILL AN AXIS SCALE BEAD        REFORM    614
                  C       GIVEN THE CURRENT AXIS POINTER.                                   REFORM    615
      5           C                                                                         REFORM    616
                  C       * FORMAT              *                                           REFORM    617
                  C       * YMIN OR MAX         *                                           REFORM    618
                  C       * X INCREMENT         *                                           REFORM    619
                  C       * Y1 AXIS MIN         *                                           REFORM    620
     10           C       * Y1 AXIS INCREMENT   *                                           REFORM    621
                  C       *     .   .   .       *                                           REFORM    622
                  C       * YN AXIS MIN         *                                           REFORM    623
                  C       * YN AXIS INCREMENT   *                                           REFORM    624
                  C       * * * * * * * * * * *                                           REFORM    625
     15           C                                                                         REFORM    626
                        COMMON /ITABLE/ IELETAB,IFUNCTB,XSCAL,YSCAL                         REFORM    627
                        COMMON /CMPCDS/ IWH0,IWH1,IWH2,IWH3,IWH4,IWH5,IWH6                  REFORM    628
                        DATA ID /6HSCL80/                                                   REFORM    629
                  C                                                                         REFORM    630
     20           C       CREATE AND FILL AXIS SCALE BEAD                                   REFORM    631
                        CALL DMGTBD(8,ISCL80)                                               MODB       12
                        IF (ISCL80 .EQ. 0) GO TO 9991                                       REFORM    633
                  C                                                                         REFORM    634
                  C       GET X AXIS INFO                                                   REFORM    635
     25                 CALL DMGET(IWH2,IAXSPT,INDX)                                      REFORM    636
                        CALL SEARCH(IELETAB, 2,INDX)                                        REFORM    637
                        IF (INDX .EQ. 0) GO TO 9992                                         REFORM    638
                        SMIN = GETVAL(IELETAB, 4,INDX)                                      REFORM    639
                        DELTA = GETVAL(IELETAB, 5,INDX)                                     REFORM    640
     30                 CALL DMSET(IWH1,ISCL80,SMIN)                                      REFORM    641
                        CALL DMSET(IWH2,ISCL80,DELTA)                                       REFORM    642
                  C                                                                         REFORM    643
                  C       GET AXIS INFO FOR EACH Y AXIS                                     REFORM    644
                        DO 10 I=1,NOFYAX                                                    REFORM    645
     35                 CALL DMGET(IWH4+I,IAXSPT,INDX)                                    REFORM    646
                        CALL SEARCH(IELETAB, 2,INDX)                                        REFORM    647
                        IF (INDX .EQ. 0) GO TO 9992                                         REFORM    648
                        SMIN = GETVAL(IELETAB, 6,INDX)                                      REFORM    649
                        DELTA = GETVAL(IELETAB, 7,INDX)                                     REFORM    650
     40                 CALL DMSET(IWH1+2*I,ISCL80,SMIN)                                  REFORM    651
                        CALL DMSET(IWH2+2*I,ISCL80,DELTA)                                   REFORM    652
                     10 CONTINUE                                                            REFORM    653
                        RETURN                                                              REFORM    654
                  C                                                                         REFORM    655
     45           C       ERROR CALLS                                                       REFORM    656
                  9991  CALL ERRORM(26H(21H BEAD SPACE EXHAUSTED),-3,ID)                   REFORM    657
                  9992  CALL ERRORM(34H(29H INTERNAL ERROR - SEE ANALYST),-3,ID)            REFORM    658
                        END                                                                 REFORM    659
```

```
      SUBROUTINE SSORT(ARRAY,NROW,NCOL,NELE,INDSRT)                    MODA   438
      DIMENSION ARRAY(NROW,NCOL)                                       MODA   439
C     M = NELE                                                         MODB    13
      DO 5 L=1,NELE,L                                                  MODA   441
    5 M = 2*L - 1                                                      MODA   442
C                                                                      MODA   443
C        START LOOP                                                    MODA   444
   10 M = M / 2                                                        MODA   445
      IF (M .EQ. 0) GO TO 50                                           MODA   446
      K = NELE - M                                                     MODB    14
      DO 40 J=1,K                                                      MODA   448
C                                                                      MODA   449
C        DIVIDE ARRAY INTO SUBSEQUENCE                                 MODA   450
C     DO 30 L=J,1,-M                                                   MODA   451
      L = J                                                            MODA   452
   15 IF (L .LT. 1) GO TO 40                                           MODA   453
      IF (ARRAY(L+M,INDSRT)  .GE. ARRAY(L,INDSRT)) GO TO 40            MODA   454
C                                                                      MODA   455
C        EXCHANGE POSITIONS IN ARRAY                                   MODA   456
      DO 20 INDX=1,NCOL                                                MODA   457
      TEMP = ARRAY(L,INDX)                                             MODA   458
      ARRAY(L,INDX) = ARRAY(L+M,INDX)                                  MODA   459
      ARRAY(L+M,INDX)= TEMP                                            MODA   460
   20 CONTINUE                                                         MODA   461
   30 CONTINUE                                                         MODA   462
      L = L - M                                                        MODA   463
      GO TO 15                                                         MODA   4
   40 CONTINUE                                                         MODA   4
      GO TO 10                                                         MODA   466
   50 RETURN                                                           MODA   467
      END                                                              MODA   468
```

```
      SUBROUTINE TRIM                                                   MODA   469
      COMMON /TRIM/ NAXES,NSTOR(3)                                      TRIM     2
C                                                                       AB     2
C        AXIS READ                                                      AB     3
      COMMON /AB/ IABNXT,IABLST,IABT,IABPM,IABXN,IABNY                  AB     4
C                                                                       PM     2
C        PARAMETER - MULTIPLIER READ                                    PM     3
      COMMON /PM/ IPMNXT,IPMLST,IPMPL,IPMML,IPMPB,IPMPT,IPMMT           PM     4
C                                                                       PB     2
C        PLOT READ                                                      PB     3
      COMMON /PB/ IPBNXT,IPBNT,IPBLST,IPBLT,IPBPG,IPBTT,IPBAB,IPBAS,    PB     4
     X            IPBPM,IPBLB,IPBCB,IPBNM                              PB     5
      COMMON /COMCDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IW MODA   474
      DIMENSION ITMP(3),ITEMP(3),ISTOR(3)                               MODA   475
      DIMENSION NAMX(3),NAMY(3),MASKX(3),MASKY(3)                       MODA   476
      DATA NAMX /0315000000000000000008,0114201001000000000008,         MODA   477
     X                                  0114201001000000000008/        MODA   478
      DATA NAMY /0314000000000000000008,0314000000000000000008,         MODA   479
     X                                  0304000000000000000008/        MODA   480
      DATA MASKX /7777000000000000000008,7777777777000000000008,       MODA   481
     X                                  7777777777000000000008/        MODA   482
      DATA MASKY /7777000000000000000008,7777000000000000000008,       MODA   483
     X                                  7777000000000000000008/        MODA   484
      DO 10 I=1,3                                                       MODA   485
      CALL DMGET(IABNY,NSTOR(I),ITMP(I))                                MODA   486
      IF (ITMP(I) .GT. 1) RETURN                                        MODA   487
      CALL DMGET(IABXN,NSTOR(I),ITMP(I))                                MODA   488
      IF ((ITMP(I) .AND. MASKX(I)) .NE. NAMX(I)) RETURN                 MODA   489
      CALL DMGET(IABNY+1,NSTOR(I),ITMP(I))                              MODA   490
      IF ((ITMP(I) .AND. MASKY(I)) .NE. NAMY(I)) RETURN                 MODA   491
      CALL DMGET(IABPM,NSTOR(I),NSTOR(I))                               MODA   492
      CALL DMGET(IPMPL,NSTOR(I),ITMP(I))                                MODA   493
      CALL DMGET(IPMML,NSTOR(I),ITEMP(I))                               MODA   494
      CALL DMGET(IPMPB,NSTOR(I),NSTOR(I))                               MODA   495
   10 CONTINUE                                                          MODA   496
      DO 40 I=1,2                                                       MODA   497
      CALL DMGET(IWHOLE2,ITMP(I),IHOLD)                                 MODA   498
      CALL DMGET(IWHOLE2,ITMP(I+1),JHOLD)                               MODA   499
      IF (IHOLD .NE. JHOLD) RETURN                                      MODA   500
      CALL DMGET(IWHOLE3,ITMP(I),IHOLD)                                 MODA   501
      CALL DMGET(IWHOLE3,ITMP(I+1), JHOLD)                              MODA   502
      CALL DMGET(IWHOLE3,IHOLD,NI)                                      MODA   503
      CALL DMGET(IWHOLE3,JHOLD,NJ)                                      MODA   504
      IF (NI .NE. NJ) RETURN                                            MODA   505
      MAX = NI + 4                                                      MODA   506
      DO 20 N = 1, MAX                                                  MODA   507
      CALL DMGET(IWHOLE0+N,IHOLD,NI)                                    MODA   508
      CALL DMGET(IWHOLE0+N,JHOLD,NJ)                                    MODA   509
      IF (NI .NE. NJ) RETURN                                            MODA   510
   20 CONTINUE                                                          MODA   511
      CALL DMGET(IWHOLE2,ITEMP(I),IHOLD)                                MODA   512
      CALL DMGET(IWHOLE2,ITEMP(I+1),JHOLD)                              MODA   513
      IF (IHOLD .NE. JHOLD) RETURN                                      MODA   514
      CALL DMGET(IWHOLE3,ITEMP(I),IHOLD)                                MODA   515
      CALL DMGET(IWHOLE3,ITEMP(I+1), JHOLD)                             MODA   516
      CALL DMGET(IWHOLE2,IHOLD,NI)                                      MODA   517
      CALL DMGET(IWHOLE2,JHOLD,NJ)                                      MODA   518
```

```
      IF (NI .NE. NJ) RETURN                                          MODA  519
      MAX = NI + 4                                                    MODA  520
60    DO 30 N=1,MAX                                                   MODA  521
      CALL DMGET(IWHOLE0+N,IHOLD,NI)                                  MODA  522
      CALL DMGET(IWHOLE0+N,JHOLD,NJ)                                  MODA  523
      IF (NI .NE. NJ) RETURN                                          MODA  524
   30 CONTINUE                                                        MODA  525
65 40 CONTINUE                                                        MODA  526
      LAST = 0                                                        MODA  527
   50 DO 60 I=1,3                                                     MODA  528
      IF (NSTOR(I) .EQ. 0) RETURN                                     MODA  529
      CALL DMGET(IP8NXT,NSTOR(I),ITEMP(I))                            MODA  530
70    IF (LAST .NE. 0) CALL DMSET(IP8NT,LAST,NSTOR(I))                MODA  531
      CALL DMSET(IP8LT,NSTOR(I),LAST)                                 MODA  532
      LAST = NSTOR(I)                                                 MODA  533
      NSTOR(I) = ITEMP(I)                                             MODA  534
   60 CONTINUE                                                        MODA  535
75    GO TO 50                                                        MODA  536
      END                                                             MODA  537
```

```
      1              OVERLAY(6,0)                                                    PLOT        2
                     SUBROUTINE AXES, RETURNS(NEXT)                                  PLOT        3
                C                                                                    PLOT        4
                C       THIS SUBROUTINE UPDATES THE AXES BEAD POINTER SO THAT IT     PLOT        5
      5         C       CONTAINS THE ADDRESS OF NEXT AXES BEAD TO PROCESS FOR A      PLOT        6
                C       PARTICULAR PLOT GROUP. IF ONE IS NOT PRESENT A RETURN NEXT IS PLOT       7
                C       TAKEN SO THAT A NEW PG BEAD CAN BE FETCHED.                  PLOT        8
                C                                                                    PLOT        9
                      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR              PLOT       10
     10               COMMON/AXESIND/ NOFYAX,NOYAX,IXINDX,IYINDX(20)                 PLOT       11
                      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                     PLOT       12
                      COMMON/STATSWT/ ISTAT,ISTATM,ISTATP,STRSWT                     PLOT       13
                      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                             PLOT       14
                      COMMON/CMPOOS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5, PLOT       15
     15               X IWHOLE6                                                      PLOT       16
                      DATA NOYAX/0/,NOFYAX/0/                                        PLOT       17
                C                                                                    PLOT       18
                C       UPDATE OLD Y AXES                                            PLOT       19
                      NOYAX = NOFYAX                                                 PLOT       20
     20               C                                                              PLOT       21
                C       CHECK WETHER INITIALIZATION OR UPDATE                        PLOT       22
                      IF (ISTAT) GO TO 30                                            PLOT       23
                C                                                                    PLOT       24
                C       UPDATE CURRENT AXES POINTER                                  PLOT       25
     25               CALL CMGET(IWHOLE1,IAXSPT,IAXSPT)                              PLOT       26
                C                                                                    PLOT       27
                C       NEXT BEAD PRESENT                                            PLOT       28
                      IF (IAXSPT .EQ. 0) RETURN NEXT                                 PLOT       29
                      ISTAT = .TRUE.                                                 PLOT       30
     30               GO TO 40                                                       PLOT       33
                C                                                                    PLOT       34
                C       INITIALIZE AXES POINTER FROM PG BEAD                         PLOT       35
                   30 CALL CMGET(IWHOLE4,IPGPT,IAXSPT)                               PLOT       36
                C                                                                    PLOT       37
     35         C     *** INITIALIZE AXES INFORMATION                                PLOT       38
                C       GET NUMBER OF Y AXES                                         MODA      538
                   40 CONTINUE                                                       MODA      539
                      CALL CMGET(IWHOLE6,IAXSPT,NOFYAX)                              MODA      540
                C                                                                    MODA      541
     40         C       CONVERT X AXIS NAME TO ELEMENT TABLE INDEX                   MODA      542
                      CALL CMGET(IWHOLE5,IAXSPT,IXINDX)                              MODA      543
                      CALL SEARCH(IELETAB,1,IXINDX)                                  MODA      544
                      IF (IXINDX .EQ. 0) GO TO 9991                                  MODA      545
                C                                                                    MODA      546
     45         C       DO SAME FOR EACH Y AXIS                                      MODA      547
                      DO 50 J=1,NOFYAX                                               MODA      548
                      CALL CMGET(IWHOLE6+J,IAXSPT,IYINDX(J))                         MODA      549
                      CALL SEARCH(IELETAB,1,IYINDX(J))                               MODA      550
                      IF (IYINDX(J) .EQ. 0) GO TO 9991                               MODA      551
     50               50 CONTINUE                                                    MODA      552
                      RETURN                                                         MODA      553
                C                                                                    PLOT       60
                C       ERROR CALLS                                                  PLOT       61
                 9991 CALL ERROR(42H(37H AXIS ELEMENT NAME NOT FOUND IN TABLE),-3,   PLOT       62
     55              X 4HAXES)                                                       PLOT       63
                      END                                                            PLOT       64
```

```
      SUBROUTINE DISAXES                                                PLOT      65
                                                                        PLOT      66
C                                                                       PLOT      67
C       THIS SUBROUTINE USES THE PLOT PACKAGE TO SCALE AND DISPLAY A    PLOT      68
C       LABELED X AND Y AXIS.                                           PLOT      69
C                                                                       PLOT      70
      COMMON/AXESIND/ NOFYAX,NOYAX,IXINDX,IYINDX(20)                    PLOT      71
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCL,YSCL                          PLOT      72
C                                                                       PLOT      73
C       RETRIEVE X AXIS INFO                                            PLOT      74
      NAME  = LOADVAL(IELETAB, 1, IXINDX)                               PLOT      75
      AMNMX = GETVAL(IELETAB, 4, IXINDX)                                PLOT      76
      DELTA = GETVAL(IELETAB, 5, IXINDX)                                PLOT      77
C                                                                       PLOT      78
C       DRAW X AXIS                                                     PLOT      79
      CALL AXIS(0.0,0.0,NAME ,-10,XSCL,0.0,AMNMX,DELTA)                 PLOT      80
C                                                                       PLOT      81
C       DISPLAY EACH Y AXIS                                             PLOT      82
      IF (NOYAX .GT. 3) GO TO 60                                        PLOT      83
      DO 50 J=1,NOFYAX                                                  PLOT      84
      IF (J.NE.1) GO TO 10                                              PLOT      85
      X = 0.0                                                           PLOT      86
      NC = 10                                                           PLOT      87
      GO TO 30                                                          PLOT      88
   10 IF (J.NE.2) GO TO 20                                              PLOT      89
      X = -.5                                                           PLOT      90
      GO TO 30                                                          PLOT      91
   20 X = 9.5                                                           PLOT      92
      NC = -10                                                          PLOT      93
C                                                                       PLOT      94
C       RETRIEVE Y AXIS INFO                                            PLOT      95
   30 NAME = LOADVAL(IELETAB, 1, IYINDX(J))                             PLOT      96
      AMNMX= GETVAL(IELETAB, 6, IYINDX(J))                              PLOT      97
      DELTA = GETVAL(IELETAB, 7, IYINDX(J))                             PLOT      98
      CALL AXIS(X,0.0,NAME,NC,YSCL,90.0,AMNMX,DELTA)                    PLOT      99
   50 CONTINUE                                                          PLOT     100
      RETURN                                                            PLOT     101
   60 X = 9.5                                                           PLOT     102
      DO 70 J=1,NOFYAX                                                  PLOT     103
      NAME = LOADVAL(IELETAB, 1, IYINDX(J))                             PLOT     104
      AMNMX = GETVAL(IELETAB, 6, IYINDX(J))                             PLOT     105
      DELTA = GETVAL(IELETAB, 7, IYINDX(J))                             PLOT     106
      CALL AXIS(X,0.0,NAME,-10,YSCL,90.0,AMNMX,DELTA)                   PLOT     107
      X = X + .5                                                        PLOT     108
   70 CONTINUE                                                          PLOT     109
      RETURN                                                            PLOT     110
      END                                                               PLOT     111
```

```
      SUBROUTINE DISAXS                                                 PLOT      112
C                                                                       PLOT      113
C        THIS SUBROUTINE DISPLAYS AXIS INFORMATION ON THE LINE PRINTER  PLOT      114
C                                                                       PLOT      115
      COMMON/AXESINC/ NOFYAX,NOYAX,IXINOX,IYINOX(20)                    PLOT      116
      COMMON /CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                MODA      554
      COMMON /CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,  MODB       15
     X IWHOLE6                                                          MODB       16
C                                                                       PLOT      118
C        GET X-AXIS NAME AND DISPLAY IT                                 PLOT      119
      CALL DMGET(IWHOLE6,IAXSPT,NAME)                                   MODA      555
      WRITE(6,10) NAME                                                  PLOT      121
   10 FORMAT(*0X AXIS = *,A10)                                          PLOT      122
C                                                                       PLOT      123
C        GET AND DISPLAY EACH Y AXIS NAME                               PLOT      124
      DO 30 I=1,NOFYAX                                                  PLOT      125
      CALL DMGET(IWHOLE6+I,IAXSPT,NAME)                                 MODA      556
      WRITE(6,20) I,NAME                                                PLOT      127
   20 FORMAT(*0Y*,I2,* AXIS = *,A10)                                    PLOT      128
   30 CONTINUE                                                          PLOT      129
      RETURN                                                            PLOT      130
      END                                                               PLOT      131
```

```
      SUBROUTINE DISCON(Y, X)                                                PLOT      132
C                                                                            PLOT      133
C        THIS SUBROUTINE DETERMINES IF THERE ARE ANY CONDITIONS TO BE        PLOT      134
C        DISPLAYED AND IF SO USES THE X,Y COORDINATES GIVEN.                 PLOT      135
C                                                                            PLOT      136
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                      PLOT      137
      COMMON/CWECDS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,         PLOT      138
     X IWHOLE6                                                               PLOT      139
      INTEGER BCDHOLD(8)                                                     PLOT      140
C                                                                            PLOT      141
C        DETERMINE IF CONDITIONS ARE PRESENT                                 PLOT      142
      IF (ICONPT .NE. 0) GO TO 20                                            PLOT      143
      CALL SYMBOL(X,Y,.105,13HNO CONDITIONS,0.0,13)                          PLOT      144
      WRITE(6,17)                                                            PLOT      145
   17 FORMAT(22H NO CONDITIONS PRESENT)                                      PLOT      146
      RETURN                                                                 PLOT      147
C                                                                            PLOT      148
C        CONDITIONS PRESENT GET TEXT                                         PLOT      149
   20 IPTR = ICONPT                                                          PLOT      150
      CALL SYMBOL(X,Y,.105,10HCONDITIONS,0.0,10)                             PLOT      151
      WRITE(6,21)                                                            PLOT      152
   21 FORMAT(11HCONDITIONS)                                                  PLOT      153
   22 CALL DMGET(07000100000C3B,IPTR,BCDHOLD)                                PLOT      154
C                                                                            PLOT      155
C        GET NUMBER OF TEXT CHARACTERS                                       PLOT      156
      CALL DMGET(IWHOLE2,IPTR,NCHAR)                                         PLOT      157
C                                                                            PLOT      158
C        ALTER X COORD                                                       PLOT      159
      Y = Y -(.105 +.05)                                                     PLOT      160
C                                                                            PLOT      161
C        DISPLAY NTH CONDITION                                               PLOT      162
      IF (X .EQ. -99.99) GO TO 24                                            PLOT      163
      CALL SYMBOL(X,Y,.105,BCDHOLD,0.0,NCHAR)                                PLOT      164
   24 CONTINUE                                                               PLOT      165
      WRITE(6,25) (BCDHOLD(I),I=1,8)                                         PLOT      166
   25 FORMAT(1H0,8A10)                                                       PLOT      167
C                                                                            PLOT      168
C        GET POINTER TO NEXT CONDITION                                       PLOT      169
      CALL DMGET(IWHOLE1,IPTR,IPTR)                                          PLOT      170
C                                                                            PLOT      171
C        CHECK IF FINISHED                                                   PLOT      172
      IF (IPTR .EQ. 0) RETURN                                                PLOT      173
      GO TO 22                                                               PLOT      174
      END                                                                    PLOT      175
```

```
      SUBROUTINE DISINFO(I,IPTR)                                        PLOT     176
C                                                                       PLOT     177
C        THIS SUBROUTINE DISPLAYS PLOTTING INFORMATION NOT ONLY ON THE  PLOT     178
C        CURRENT PLOT, BUT ALSO ON THE PRINTER.                         PLOT     179
C                                                                       PLOT     180
C        I     - NUMBER OF PARAMETERS ON CURRENT PLOT                   PLOT     181
C                IF NEGATIVE, INFORMATION WILL NOT FIT ON PLOT          PLOT     182
C                IF ZERO, NO PLOT WAS GENERATED                         PLOT     183
C        IPTR - POINTER TO BEGINNING OF PLOT HEAD INFO                  PLOT     184
C                                                                       PLOT     185
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT     186
      COMMON/CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT     187
     X IWHOLE6                                                          PLOT     188
      COMMON/STATS/ITOTPTS,ITOTDEL,NTPBL,NUMINPG,ITOTPLT                PLOT     189
      COMMON/AXESINC/ NOFYAX,NOYAX,IXINDX,IYINDX(20)                    PLOT     190
C                                                                       PLOT     191
C        CHECK IF LABELS WILL FIT ON PLOT                               PLOT     192
      IF (I) 15,5,10                                                    PLOT     193
C                                                                       PLOT     194
C        CALL NOPTS ROUTINE                                             PLOT     195
    5 CALL NOPTS                                                        PLOT     196
      RETURN                                                            PLOT     197
C                                                                       PLOT     198
C     ***  PUT INFO ON PLOT                                             PLOT     199
C                                                                       PLOT     200
C        ASSIGN CO-ORDS FOR MIDDLE OF TITLE                             PLOT     201
   10 CONTINUE                                                          PLOT     202
      HT = 6.0                                                          PLOT     203
      WT=4.5                                                            PLOT     204
      CALL DISTEXT(HT,WT,.140)                                          PLOT     205
C                                                                       PLOT     206
C        COMPUTE COORDINATES FOR CONDITIONS TO START                    PLOT     207
      HT = 8.5 -(.105 + .05 + .5)                                       PLOT     208
      WT = .1 -1.0                                                      PLOT     209
C                                                                       PLOT     210
C        DISPLAY CONDITIONS, IF ANY                                     PLOT     211
      CALL DISCON( HT, WT)                                              PLOT     212
C                                                                       PLOT     213
C        COMPUTE COORD FOR P-M TO START                                 PLOT     214
      HT = 8.5 -(.105 + .05 + .5)                                       PLOT     215
      WT = 11. -(30. * .1 +1.0)                                         PLOT     216
C                                                                       PLOT     217
C        DISPLAY P-M INFO, IF ANY                                       PLOT     218
      CALL DISPM(HT,WT,I,IPTR)                                          PLOT     219
      GO TO 30                                                          PLOT     220
C                                                                       PLOT     221
C        SET NEW ORIGIN                                                 PLOT     222
   15 NTPBL = NTPBL +1                                                  PLOT     223
      XDIST = 12.0                                                      PLOT     224
      IF (NOYAX .GT. 3) XDIST = 11.5 + FLOAT(NOYAX) * .5                PLOT     225
      CALL PLOT(XDIST,0.0,-23)                                          PLOT     226
      CALL FRAME                                                        PLOT     227
C                                                                       PLOT     228
C        DISPLAY TITLE INFO                                             PLOT     229
      HT = 8.5 - (.105 + .05 + .5)                                      PLOT     230
      WT = (11. -1.0) / 2.                                              PLOT     231
      CALL DISTEXT(HT,WT,.105)                                          PLOT     232
```

```
      C                                                                 PLOT  233
      C        COMPUTE COORDS FOR CONDITIONS                            PLOT  234
60              HT = 8.5 -(.105 + .05 + .5)                              PLOT  235
                WT = .1 -1.0                                             PLOT  236
      C                                                                  PLOT  237
      C        DISPLAY CONDITIONS                                       PLOT  238
                CALL DISCON( HT,WT)                                      PLOT  239
65    C                                                                  PLOT  240
      C        GET NUMBER OF CONDITIONS IN PG                           PLOT  241
                CALL DMGET(IWHOLE6,IPGPT,J)                              PLOT  242
      C                                                                  PLOT  243
      C        COMPUTE CO-ORDS FOR P-M                                  PLOT  244
70              HT = 8.5 - ((.105+.05) * (J+2) + .5)                     PLOT  245
                WT = .1 -1.0                                             PLOT  246
                I = -I                                                   PLOT  247
      C                                                                  PLOT  248
      C        DISPLAY P-M INFO                                         PLOT  249
75              CALL DISPM(HT,WT,I,IPTR)                                 PLOT  250
      C                                                                  PLOT  251
      C        WRITE OUT AXES INFO                                      PLOT  252
           30 CALL DISAXS                                                PLOT  253
      C                                                                  PLOT  254
80    C        WRITE OUT STATISTICS                                     PLOT  255
                CALL DISTAT                                              PLOT  256
                RETURN                                                   PLOT  257
                END                                                      PLOT  258
```

```
      SUBROUTINE DISL8L(Y,X)                                            PLOT      259
C                                                                       PLOT      260
C        THIS SUBROUTINE DISPLAYS THE MAX AND MIN VALUES FOR ANY        PLOT      261
C        DATA ELEMENTS WHICH WERE SPECIFIED AS LABLELS                  PLOT      262
C                                                                       PLOT      263
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT      264
      COMMON/CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT      265
     X IWHOLE6                                                          PLOT      266
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                        PLOT      267
      DIMENSION IHOLD(3)                                                PLOT      268
C                                                                       PLOT      269
C        GET POINTER TO FIRST LBL READ                                  PLOT      270
      CALL DMGET(IWHOLE0+10,IPLTPTR,LBLPTR)                             PLOT      271
    5 IF (LBLPTR .EQ. 0) RETURN                                         PLOT      272
C                                                                       PLOT      273
C        GET LABEL NAME                                                 PLOT      274
      CALL DMGET(IWHOLE2,LBLPTR,LNAME)                                  PLOT      275
      CALL SEARCH(IELETAB,2,LNAME)                                      PLOT      276
      IF (LNAME .EQ. 0) GO TO 9991                                      PLOT      277
      LNAME = LOADVAL(IELETAB,1,LNAME)                                  PLOT      278
C                                                                       PLOT      279
C        GET LABEL MAX AND MIN VALUES                                   PLOT      280
      CALL DMGET(IWHOLE3,LBLPTR,VAL)                                    PLOT      281
      INAME = 6H MAX =                                                  PLOT      282
C                                                                       PLOT      283
C        DISPLAY LABEL INFO                                             PLOT      284
   10 ENCODE(30,20,IHOLD) LNAME,INAME,VAL                               PLOT      285
   20 FORMAT(A10,A6,G12.6,2X)                                           PLOT      286
      CALL SYMBOL(X,Y,.105,IHOLD,0.0,28)                                PLOT      287
      Y = Y - (.105 + .05)                                              PLOT      288
      WRITE(6,30) (IHOLD(K),K=1,3)                                      PLOT      289
   30 FORMAT(1HC,3A10)                                                  PLOT      290
      IF (LNAME .EQ. 1H ) GO TO 40                                      PLOT      291
C                                                                       PLOT      292
C        PREPARE MAX VALUE                                              PLOT      293
      LNAME = 1H                                                        PLOT      294
      INAME = 6H MIN =                                                  PLOT      295
      CALL DMGET(IWHOLE4,LBLPTR,VAL)                                    PLOT      296
      GO TO 10                                                          PLOT      297
C                                                                       PLOT      298
C        UPDATE LABEL POINTER                                           PLOT      299
   40 CALL DMGET(IWHOLE1,LBLPTR,LBLPTR)                                 PLOT      300
      GO TO 5                                                           PLOT      301
C                                                                       PLOT      302
C        ERROR CALLS                                                    PLOT      303
 9991 CALL ERRORM(33H(28H LABEL NO NOT FOUND IN TABLE),-3,6HDISL8L)      PLOT      304
      END                                                               PLOT      305
```

```
      SUBROUTINE DISPM(Y,X,I,IPTR)                                      PLOT    306
C                                                                       PLOT    307
C        THIS SUBROUTINE DETERMINES WHETHER ANY INFORMATION EXISTS AND  PLOT    308
C        IF SO PLOTS IT WITH THE X,Y COORDS GIVEN AND DISPLAYS IT ON THE PLOT    309
C        LINE PRINTER.                                                  PLOT    310
C                                                                       PLOT    311
      COMMON/STATSWT/ ISTAT,ISTATM,ISTATP,STRSWT                        PLOT    312
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                PLOT    313
      COMMON/ITABLE/ IFLETAB,IFUNTAB,XSCAL,YSCAL                        PLOT    314
      COMMON/AXESING/ NOFYAX,NOYAX,IOX,IOY(20)                          PLOT    315
      COMMON/PMNUMBR/ MNUMBS,NUMBP                                      PLOT    316
      COMMON/PMTEXT/ LENGTH,MNAMES,MSTRUCT,MTEXT,NAMEP                  PLOT    317
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT    318
      COMMON/CMRCDS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,    PLOT    319
     X IWHOLE6                                                          PLOT    320
      DIMENSION IHOLD(3)                                                PLOT    321
      IF (ISTATP) GO TO 4                                               PLOT    322
      CALL SYMBOL(X,Y,.105,19HNO PARAMETER VALUES, 0.0,19)              PLOT    323
      WRITE(6,1)                                                        PLOT    324
    1 FORMAT(20HNO PARAMETER VALUES)                                    PLOT    325
      Y = Y -(.105 + .05)                                               PLOT    326
      IF (NOFYAX .NE. 1) CALL MLTAXS(Y,X)                               PLOT    327
      GO TO 30                                                          PLOT    328
C                                                                       PLOT    329
C     ***  DISPLAY PARAMETER INFORMATION                                PLOT    330
    4 DO 20 J=1,I                                                       PLOT    331
C                                                                       PLOT    332
C        GET PARAMETER VALUE                                            PLOT    333
      CALL DMGET(IWHOLE4,IPTR,PVAL)                                     MODA    557
C                                                                       MODA    558
C        GET PARAMETER NAME                                             MODA    559
      CALL DMGET(IWHOLE3,IPTR,NAMEP)                                    MODA    560
C                                                                       MODA    561
C        CREATE PARAMETER TEXT                                          MODA    562
      CALL TEXT(1HP,J,NAMEP,PVAL,IHOLD)                                 MODB     17
      X1 = X +(13.5 * .105)                                             PLOT    353
      Y1 = Y +(.5 *.105)                                                PLOT    354
C                                                                       PLOT    355
C        GET PARAMETER SYMBOL                                           PLOT    356
      CALL DMGET(IWHOLE6,IPTR,K)                                        MODA    564
      CALL SYMBOL(X1,Y1,.105,K,0.0,-1)                                  PLOT    358
      WRITE(6,10)(IHOLD(K),K=1,3)                                       PLOT    359
   10 FORMAT(1H0,19X,3A10)                                              PLOT    360
      Y = Y -(.105 +.05)                                                PLOT    361
C                                                                       PLOT    362
C        UPDATE POINTER TO PLOT HEAD                                    PLOT    363
      CALL DMGET(IWHOLE1,IPTR,IPTR)                                     PLOT    364
   20 CONTINUE                                                          PLOT    365
C                                                                       PLOT    366
C     ***  DISPLAY MULTIPLIER INFORMATION                               PLOT    367
   30 IF (ISTATM) GO TO 35                                              PLOT    368
      CALL SYMBOL(X,Y,.105,20HNO MULTIPLIER VALUES,0.0,20)              PLOT    369
      Y = Y -(.105+.05)                                                 PLOT    370
      WRITE(6,33)                                                       PLOT    371
   33 FORMAT(21HONO MULTIPLIER VALUES)                                  PLOT    372
      GO TO 72                                                          PLOT    373
   35 LTMPLT = 30                                                       PLOT    374
```

```
                LIMMLT =  30                                                    PLOT     375
                NOMLT = LENGTH                                                  PLOT     376
60              IF (LENGTH .GT. LIMMLT) NOMLT = LIMMLT                          PLOT     377
                DO 40 J=1,NOMLT                                                 PLOT     378
                CALL DMGET(IWHOLE0+12+J,IPLTPTR,NAME)                           MODA     565
                CALL DMGET(IWHOLE0+13+J,IPLTPTR,VAL)                            MODA     566
                CALL TEXT(1HM,J,NAME , VAL,IHOLD)                               MODB      18
65              CALL SYMBOL(X,Y,.105,IHOLD,0.0,28)                              PLOT     381
                WRITE(6,10) (IHOLD(I),I=1,3)                                    PLOT     382
                Y = Y -(.105 +.05)                                              PLOT     383
             40 CONTINUE                                                        PLOT     384
           C                                                                    PLOT     385
70         C       DISPLAY LABEL INFO                                           PLOT     386
             70 CALL DISLBL(Y,X)                                                PLOT     387
                RETURN                                                          PLOT     388
           C                                                                    PLOT     389
           C       ERROR CALLS                                                  PLOT     390
75          9991 CALL ERRORM(37H(32H PARAMETER NO NOT FOUND IN TABLE),-3,5HDISPM)PLOT     391
                END                                                             PLOT     392
```

```
      SUBROUTINE DISPTS(IPTR)                                                PLOT     393
C                                                                            PLOT     394
C        THIS SUBROUTINE PLOTS ALL OF THE POINTS ASSOCIATED WITH A           PLOT     395
C        PARTICULAR PARAMETER BEAD, DELETING ANY WHICH FALL OUTSIDE OF       PLOT     396
C        THE RANGE OF THE X AND Y AXES.                                      PLOT     397
C                                                                            PLOT     398
C        IPTR - POINTER TO A PARAMETER BEAD                                  PLOT     399
C        IX   - INDEX TO X-AXIS INFORMATION                                  PLOT     400
C        IY   - INDEX TO Y-AXIS INFORMATION                                  PLOT     401
C        I    - PARAMETER NUMBER FOR A PARTICULAR PLOT                       PLOT     402
C                                                                            PLOT     403
      COMMON/STATS/ ITOTPTS,ITOTDEL,NTPBL,NUMINPG,ITOTPLT                    PLOT     404
      COMMON/AXESIND/ N,NO,IX,IY(20)                                         PLOT     405
      COMMON/CMPCOS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,         PLOT     406
     X IWHOLE6                                                               PLOT     407
      COMMON /ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                            PLOT     408
      COMMON/LIMIT/ XMIN,XMAX,YMIN,YMAX                                      PLOT     409
      COMMON /INITSW/ LISTSF,SWITCH,NOOFTP                                   PLOT     410
      LOGICAL SWITCH                                                         PLOT     411
      DIMENSION XPTS(52),YPTS(52),RECNO(50)                                  PLOT     412
      COMMON/PTBDSZ/ NPTIN3D                                                 PLOT     413
      DATA NUMLEFT/0/                                                        PLOT     414
C                                                                            PLOT     415
C        COMPUTE THE LIMITS OF THE X AXIS SCALE                              PLOT     416
      CALL LIMITS(IX,1,XMAX,XMIN)                                            PLOT     417
C                                                                            PLOT     418
C        GET THE PARAMETER SYMBOL NUMBER                                     PLOT     419
      CALL DMGET(IWHOLE4,IPTR,I)                                             PLOT     420
      J = 1                                                                  MODB      19
C                                                                            PLOT     421
C        GET POINTER TO X POINT BEAD                                         PLOT     425
      CALL DMGET(IWHOLE0+7,IPTR,IRECPT)                                      MODA     568
      CALL DMGET(IWHOLE0+8,IPTR,IXPTS)                                       MODA     569
C                                                                            PLOT     428
C        COMPUTE THE LIMITS OF THE Y AXIS SCALE                              PLOT     429
      CALL LIMITS(IY(J),2,YMAX,YMIN)                                         PLOT     430
C                                                                            PLOT     431
C        WRITE OUT HEADER FOR JTH Y AXIS                                     PLOT     432
      IF (SWITCH) WRITE(6,1)                                                 PLOT     433
    1 FORMAT(1HO,9X,1HX,15X,1HY,19X,13HRECORD NUMBER)                        PLOT     434
C                                                                            PLOT     435
C        GET POINTER TO Y BEAD                                               PLOT     436
      CALL DMGET(IWHOLE0+9,IPTR,IYPTS)                                       MODA     570
C                                                                            PLOT     438
C        GET THE NUMBER OF POINTS ASSOCIATED WITH PLOT BEAD                  PLOT     439
      CALL DMGET(IWHOLE5,IPTR,NOFPTS)                                        MODA     571
C                                                                            PLOT     441
C        INCREMENT TOTAL POINT COUNTER                                       PLOT     442
      ITOTPTS = ITOTPTS + NOFPTS                                             PLOT     443
      MAXPTS = 249                                                           MODB      20
      NUMLEFT = 0                                                            MODB      21
      IF (NOFPTS .LE. MAXPTS) GO TO 25                                       MODB      22
      NUMLEFT = NOFPTS - MAXPTS                                              MODB      23
      NOFPTS = MAXPTS                                                        MODB      24
C                                                                            PLOT     447
C        FILL X AND Y POINT ARRAYS                                           PLOT     448
   25 ICOMP = 2700000000000013 + SHIFT(NOFPTS,18)                            PLOT     449
```

```
                CALL DMGET(ICOMP,IXPTS,XPTS)                                    PLOT     450
                CALL DMGET(ICOMP,IYPTS,YPTS)                                    PLOT     451
   60           CALL DMGET(ICOMP,IRECPT,RECNO)                                  PLOT     452
              C                                                                 PLOT     453
              C     IF REQUESTED PRINT POINTS RETRIEVED FROM BEADS              PLOT     454
                IF(SWITCH) CALL PRNTPTS(XPTS,YPTS,RECNO,NOFPTS)                 PLOT     455
              C                                                                 PLOT     456
   65         C     DELETE PAIRS OF POINTS OUT OF RANGE                         PLOT     457
                NOFDEL = NOFPTS                                                 PLOT     458
                CALL RANGE(XPTS,YPTS,RECNO,NOFDEL)                              MODB      25
              C                                                                 PLOT     460
              C     INCREMENT TOTAL NUMBER OF DELETED POINTS                    PLOT     461
   70           TTOTDEL = TTOTDEL + NOFDEL                                      PLOT     462
              C                                                                 PLOT     463
              C     COMPUTE NUMBER OF POINTS TO BE PLOTTED                      PLOT     464
                NOFPTS = NOFPTS -NOFDEL                                         PLOT     465
              C                                                                 MODB      26
   75         C     SAVE ONLY POINTS IN RANGE (IF GRAPHICS)                     MODB      27
                CALL DMSET(ICOMP,IXPTS,XPTS)                                    MODB      28
                CALL DMSET(ICOMP,IYPTS,YPTS)                                    MODB      29
                CALL DMSET(ICOMP,IRECPT,RECNO)                                  MODB      30
                CALL DMSET(IWHOLES,IPTR,NOFPTS)                                 MODB      31
   80           IF (NOFPTS .EQ. 0) GO TO 30                                     PLOT     466
              C                                                                 PLOT     467
              C     FILL POINT ARRAYS WITH PLOTTING INFO                        PLOT     468
                XPTS(NOFPTS+1) = GETVAL(IELETAB, 4, IX)                         PLOT     469
                XPTS(NOFPTS+2) = GETVAL(IELETAB, 5, IX)                         PLOT     470
   85           YPTS(NOFPTS+1) = GETVAL(IELETAB, 6, IY(J))                      PLOT     471
                YPTS(NOFPTS+2) = GETVAL(IELETAB, 7, IY(J))                      PLOT     472
              C                                                                 PLOT     473
              C     DISPLAY POINTS                                              PLOT     474
                K = I+J-1                                                       PLOT     475
   90           CALL LINE(XPTS,YPTS,NOFPTS,1,-1,K)                              PLOT     476
              C                                                                 PLOT     477
              C     CHECK IF MORE POINTS ARE TO BE PLOTTED                      PLOT     478
             30 IF (NUMLEFT .EQ. 0) GO TO 60                                    PLOT     479
              C                                                                 PLOT     480
   95         C     GET NEW X AND Y POINTERS                                    PLOT     481
                ICOMP = IWHOLE1 + NPTIN3D                                       PLOT     482
                CALL DMGET(ICOMP,IXPTS,IXPTS)                                   PLOT     483
                CALL DMGET(ICOMP,IYPTS,IYPTS)                                   PLOT     484
                CALL DMGET(ICOMP,IRECPT,IRECPT)                                 PLOT     485
  100           IF (NUMLEFT .GT. MAXPTS) GO TO 40                               MODB      32
                NOFPTS = MAXPTS                                                 MODB      33
                NUMLEFT = 0                                                     PLOT     488
                GO TO 50                                                        MODB      34
             40 NUMLEFT = NUMLEFT - MAXPTS                                      MODB      35
  105           NOFPTS = MAXPTS                                                 MODB      36
             50 IF (SWITCH) WRITE(6,55)                                         MODB      37
             55 FORMAT(*0THE REMAINING POINTS ARE UNAVAILABLE FOR GRAPHICS *,    MODB      38
               X *PROCESSING*)                                                  MODB      39
             60 CONTINUE                                                        PLOT     493
  110           RETURN                                                          PLOT     494
                END                                                             PLOT     495
```

```
      SUBROUTINE DISTAT                                                 PLOT      496
C                                                                       PLOT      497
C        THIS SUBROUTINE PRINTS OUT STATISTICAL INFORMATION FOR A GIVEN PLOT      498
C        PLOT.                                                          PLOT      499
C                                                                       PLOT      500
      COMMON/STATS/ITOTPTS,ITOTDEL,NTPBL,NUMINPG,ITOTPLT                PLOT      501
      WRITE(6,10) ITOTPTS                                               PLOT      502
   10 FORMAT(23HCTOTAL POINTS IN PLOT =,I6)                             PLOT      503
      WRITE(6,20) ITOTDEL                                               PLOT      504
   20 FORMAT(24HONUMBER OF POINTS NOT SHOWN =,I6)                       PLOT      505
      WRITE(6,30) NTPBL                                                 PLOT      506
   30 FORMAT(31HONUMBER OF CURRENT TAPE BLOCK =,I8)                     PLOT      507
      ITOTPTS = ITOTDEL = 0                                             PLOT      508
      RETURN                                                            PLOT      509
      END                                                               PLOT      510
```

```
      SUBROUTINE DISTEXT(Y,X,CHARHT)                                    PLOT      511
C                                                                       PLOT      512
C        GIVEN COORDINATES THIS SUBROUTINE ACCESSES TITLE INFORMATION   PLOT      513
C        STARING FROM THE SECOND WORD OF THE PLOT GROUP BEAD AND DISPLAYS PLOT     514
C        THIS INFORMATION ON BOTH THE PLOTTER AND LINE PRINTER.         PLOT      515
C                                                                       PLOT      516
      COMMON/CMRCDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT      517
     X IWHOLE6                                                          PLOT      518
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT      519
      DIMENSION IHOLD(8)                                                PLOT      520
      ICONTL = 1                                                        PLOT      521
C                                                                       PLOT      522
C        GET POINTER TO FIRST TITLE BEAD                                PLOT      523
      CALL DMGET(IWHOLE2,IPGPT,IPTR)                                    PLOT      524
   10 IF (IPTR .EQ. 0) RETURN                                           PLOT      525
C                                                                       PLOT      526
C        GET NUMBER OF CHARACTERS IN TITLE BEAD                         PLOT      527
      CALL DMGET(IWHOLE2,IPTR,NCHAR)                                    PLOT      528
C                                                                       PLOT      529
C        FILL TEXT ARRAY                                                PLOT      530
      CALL DMGET(07000100000038,IPTR,IHOLD)                             PLOT      531
C                                                                       PLOT      532
C        COMPUTE NEW X COORDINATE                                       PLOT      533
      HALF  = FLOAT(NCHAR/2)                                            PLOT      534
      XNEW = X -(HALF* CHARHT)                                          PLOT      535
C                                                                       PLOT      536
C   ***  DISPLAY TEXT                                                   PLOT      537
      IF (X .EQ. -99.99) GO TO 15                                       PLOT      5
      CALL SYMBOL(XNEW, Y,CHARHT,IHOLD,0.0,NCHAR)                       PLOT      5
   15 CONTINUE                                                          PLOT      540
      NWORD = NCHAR/10                                                  PLOT      541
      IF (MOD(NCHAR,10) .NE. 0) NWORD = NWORD +1                        PLOT      542
      WRITE(6,20) ICONTL,(IHOLD(I),I=1,NWORD)                           PLOT      543
   20 FORMAT(I1,8A10)                                                   PLOT      544
      ICONTL = 0                                                        PLOT      545
C                                                                       PLOT      546
C        ADJUST Y COORDINATE                                            PLOT      547
      Y = Y -(CHARHT + .055)                                            PLOT      548
C                                                                       PLOT      549
C        GET NEXT TITLE BEAD                                            PLOT      550
      CALL DMGET(IWHOLE1,IPTR,IPTR)                                     PLOT      551
      GO TO 10                                                          PLOT      552
      END                                                               PLOT      553
```

```
1                SUBROUTINE DISTOT1(IVAL)                                PLOT      554
                 WRITE(6,10)IVAL                                         PLOT      555
              10 FORMAT(29H0TOTAL PLOTS IN PLOT GROUP = ,I3)              PLOT      556
                 IVAL = 0                                                PLOT      557
5                RETURN                                                  PLOT      558
                 END                                                     PLOT      559
```

```
1                SUBROUTINE DISTOT2(IVAL)                                  PLOT  560
                 WRITE(6,10) IVAL                                          PLOT  561
              10 FORMAT(32H0TOTAL PLOTS GENERATED IN RUN = ,I5)             PLOT  562
                 RETURN                                                    PLOT  563
5                END                                                       PLOT  564
```

```
      SUBROUTINE FRAME                                                  PLOT    565
      COMMON/AXESINO/ NOFYAX,NOYAX,IXINOX,IYINOX(28)                    PLOT    566
      CALL PLOT(-1.0,-.5,3)                                             PLOT    567
      CALL PLOT(-1.0,8.0,2)                                             PLOT    568
      XINC = 10.0                                                       PLOT    569
      IF (NOYAX .GT. 3) XINC =  .5 * FLOAT(NOFYAX) + 9.5                PLOT    570
      CALL PLOT(XINC,8.0,2)                                             PLOT    571
      CALL PLOT(XINC,-.5,2)                                             PLOT    572
      CALL PLOT(-1.0,-.5,2)                                             PLOT    573
      RETURN                                                            PLOT    574
      END                                                               PLOT    575
```

```
1                REAL FUNCTION GETVAL(ITABPTR,ITABCOL,ITABROW)                  PLOT     576
                 EQUIVALENCE (N,GETV)                                           PLOT     577
                 N = LOADVAL(ITABPTR,ITABCOL,ITABROW)                           PLOT     578
                 GETVAL=GETV                                                    PLOT     579
5                RETURN.                                                        PLOT     580
                 END                                                            PLOT     581
```

```
      SUBROUTINE LIMITS(INDEX,ITYPE,VMAX,VMIN)                          PLOT      582
C                                                                       PLOT      583
C        THIS SUBROUTINE COMPUTES A MAXIMUM AND MINIMUM LIMIT FOR THE   PLOT      584
C        RANGE OF EITHER THE X OR Y AXIS.                               PLOT      585
C                                                                       PLOT      586
C        INDEX - INDEX TO AXIS INFORMATION                              PLOT      587
C        ITYPE - VALUE = 1  FOR X AXIS                                  PLOT      588
C                VALUE = 2  FOR Y AXIS                                  PLOT      589
C        VMAX  - MAXIMUM LIMIT                                          PLOT      590
C        VMIN  - MINIMUM LIMIT                                          PLOT      591
C                                                                       PLOT      592
      COMMON/AXESINC/ NOFYAX,NOYAX,IX,IY(20)                            PLOT      593
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCL,YSCL                          PLOT      594
C                                                                       PLOT      595
C          GET THE MIN/MAX AXES VALUE AND INCREMENT                     PLOT      596
      ICOL = 4                                                          PLOT      597
      IF (ITYPE .EQ. 2) ICOL = 6                                        PLOT      598
      VBASE = GETVAL(IELETAB, ICOL, INDEX)                              PLOT      599
      VINC = GETVAL(IELETAB, ICOL+1, INDEX)                             PLOT      600
      GO TO (10,20),ITYPE                                               PLOT      601
   10 VINC = VINC*XSCL                                                  PLOT      602
      GO TO 30                                                          PLOT      603
   20 VINC = VINC*YSCL                                                  PLOT      604
C                                                                       PLOT      605
C        DETERMINE IF BASE IS MAX OR MIN                                PLOT      606
   30 IF (VINC .GE. 0.0) GO TO 50                                       PLOT      607
C                                                                       PLOT      608
C        BASE IS MAX                                                    PLOT      609
      VMAX = VBASE + 1.E-06 * ABS(VBASE)                                PLOT      610
      VMIN = VBASE + VINC -1.E-05 * ABS(VBASE + VINC)                   PLOT      611
      RETURN                                                            PLOT      612
C                                                                       PLOT      613
C        BASE IS MIN                                                    PLOT      614
   50 VMIN = VBASE -1.E-06 * ABS(VBASE)                                 PLOT      615
      VMAX = VBASE+VINC + 1.E-06 * ABS(VBASE + VINC)                    PLOT      616
      RETURN                                                            PLOT      617
      END                                                               PLOT      618
```

```
      INTEGER FUNCTION LOADVAL(ITABPTR,ITABCOL,ITABROW)                  PLOT      619
      COMMON /CMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT      620
     X IWHOLE6                                                           PLOT      621
C                                                                        PLOT      622
C        GET POINTER TO CORRECT TABLE COL                                PLOT      623
      CALL IMGET(IWHOLE2+ITABCOL,ITABPTR,ICOLPTR)                        PLOT      624
C                                                                        PLOT      625
C        GET VALUE FROM SELECTED ROW                                     PLOT      626
      CALL IMGET(IWHOLE0+ITABROW,ICOLPTR,LOADVAL)                        PLOT      627
      RETURN                                                             PLOT      628
      END                                                                PLOT      629
```

```
      SUBROUTINE MLTAXS( Y,X)                                          PLOT      630
C                                                                      PLOT      631
C        THIS SUBROUTINE IDENTIFIES POINTS IN A MULTIPLE AXES PLOT     PLOT      632
C                                                                      PLOT      633
      COMMON/AXESIND/ NOFYAX,NOYAX,IX,IY(20)                           PLOT      634
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCL,YSCL                         PLOT      635
      DIMENSION IHOLD(2)                                               PLOT      636
C                                                                      PLOT      637
C        DO FOR EACH Y AXIS                                            PLOT      638
      DO 50 J=1,  NOFYAX                                               PLOT      639
      NAMEY = LOADVAL(IELETAB, 1, IY(J))                               PLOT      640
      ENCODE(20,10,IHOLD) J,NAMEY                                      PLOT      641
   10 FORMAT(1HA,T1,1X,A10,3H  =,4X)                                   PLOT      642
C                                                                      PLOT      643
C        DISPLAY INFO ON PLOTTER                                       PLOT      644
      CALL SYMBOL(X,Y,.105,IHOLD,0.0,16)                               PLOT      645
      X1 =  X + (18.5 * .105)                                          PLOT      646
      Y1 =  Y + (.5 * .105)                                            PLOT      647
      CALL SYMBOL(X1,Y1,.105, J,0.0,-1)                                PLOT      648
      Y = Y -(.105 +.05)                                               PLOT      649
   50 CONTINUE                                                         PLOT      650
      RETURN                                                           PLOT      651
      END                                                              PLOT      652
```

```
      SUBROUTINE PLOT80, RETURNS(NEXT)                                  MODA  572
C                                                                       PLOT  654
C     THIS SUBROUTINE UPDATES THE MULTIPLIER STRUCTURE HEAD FROM THE    PLOT  655
C     LOWEST LEVEL UPWARD TO CONTAIN THE ADDRESS REPRESENTATION OF      PLOT  656
C     SEQUENTIAL MOVEMENT TO THE NEXT PARAMETER VALUE HEAD. WHEN THIS   PLOT  657
C     REACHES ITS LIMIT A RETURN NEXT IS TAKEN.                         PLOT  658
C                                                                       PLOT  659
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                         PLOT  660
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                PLOT  661
      COMMON/CVECTPOS/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                PLOT  662
      COMMON/PRTEXT/LENGTH,MNAMES,MSTRUCT,MTEXT,NAMEP                   PLOT  663
      COMMON/PMNUMBS/ MNUMBS,NUMBP                                      PLOT  664
      COMMON/CHPOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,    PLOT  665
     X IWHOLE6                                                          PLOT  666
      DATA ID /6HPLOT80/                                                MODA  573
C                                                                       PLOT  669
C        UPDATE OR INITIALIZATION                                       PLOT  670
      IF (ISTAT) GO TO 30                                               PLOT  671
C                                                                       MODA  574
C        UPDATE NEXT PLOT HEAD                                          MODA  575
      CALL DMGET(IWHOLE1,IPLTPTR,IPLTPTR)                               MODA  576
      IF (IPLTPTR .EQ. 0) RETURN NEXT                                   MODA  577
      ISTAT = .TRUE.                                                    MODA  578
      RETURN                                                            MODA  579
C                                                                       MODA  580
C        INITIALIZE PLOT HEAD                                           MODA  581
   30 CALL DMGET(IWHOLE5,IPMPT,IPLTPTR)                                 MODA  582
C                                                                       MODA  583
C        SET P-M STATUS SWITCHES                                        MODA  584
      CALL DMGET(IWHOLE3,IPMPT,IVAL)                                    MODA  585
      IF (IVAL .NE. 0) ISTATP = .TRUE.                                  MODA  586
      CALL DMGET(IWHOLE4,IPMPT,IVAL)                                    MODA  587
      IF (IVAL .NE. 0) ISTATM = .TRUE.                                  MODA  588
      IF (IPLTPTR .EQ. 0) GO TO 40                                      MODA  589
      STRSWT = .TRUE.                                                   MODA  590
      CALL DMGET(IWHOLE0+12,IPLTPTR,LENGTH)                             MODA  591
   40 RETURN                                                            MODA  592
      END                                                               PLOT  759
```

```
      SUBROUTINE NOPTS                                                  PLOT  760
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT  761
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCAL,YSCAL                        PLOT  762
      COMMON/CMPOOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT  763
     X IWHOLE6                                                          PLOT  764
      COMMON/PMTEXT/LENGTH,MNAMES,MSTRUCT,MTEXT,NAMEP                   PLOT  765
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                         PLOT  766
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                PLOT  767
      DIMENSION IHOLD(2)                                                PLOT  768
      X = -99.99                                                        PLOT  769
      Y = 0.0                                                           PLOT  770
      CALL DISTEXT(X,Y,Y)                                               PLOT  771
      CALL DISCON(X ,Y)                                                 PLOT  772
      WRITE(6,10)                                                       PLOT  773
   10 FORMAT(20HONO POINTS GENERATED)                                   PLOT  774
      IF (ISTATP) GO TO 30                                              PLOT  775
C                                                                       PLOT  776
C        CHECK FOR PARAMETER NAME                                       PLOT  777
      WRITE(6,20)                                                       PLOT  778
   20 FORMAT(19HONO PARAMETER NAMES)                                    PLOT  779
      GO TO 50                                                          PLOT  780
C                                                                       PLOT  781
C         GET FIRST PARAMETER NUMBER                                    PLOT  782
   30 CALL DMGET(IWHOLE2,IPMPT,IPTR)                                    PLOT  783
C                                                                       PLOT  784
C        SAVE FIRST PTR                                                 PLOT  785
      IOPTR = IPTR                                                      PLOT  786
   33 CALL DMGET(IWHOLE2,IPTR,NAMEP)                                    PLOT  787
      ENCODE(20,35,IHOLD) NAMEP                                         PLOT  788
   35 FORMAT(9HP NAME = ,A10,1X)                                        PLOT  794
C                                                                       PLOT  795
C        DISPLAY PARAMETER NAME                                         PLOT  796
      WRITE(6,40) (IHOLD(K),K=1,2)                                      PLOT  797
   40 FORMAT(1H0,2A10)                                                  PLOT  798
C                                                                       PLOT  799
C        UPDATE PARAMETER LIST POINTER                                  PLOT  800
      CALL DMGET(IWHOLE1,IPTR,IPTR)                                     PLOT  801
      IF (IPTR .NE. 0) GO TO 33                                         PLOT  802
C                                                                       PLOT  803
C        CHECK FOR MULTIPLIER NAMES                                     PLOT  804
   50 IF (ISTATM) GO TO 70                                              PLOT  805
      WRITE(6,60)                                                       PLOT  806
   60 FORMAT(20HONO MULTIPLIER NAMES)                                   PLOT  807
      GO TO 95                                                          PLOT  808
C                                                                       PLOT  809
C        DISPLAY MULTIPLIER NAMES                                       PLOT  810
   70 LIMMLT = 20                                                       PLOT  811
      NOMLT = LENGTH                                                    PLOT  812
      IF (LENGTH .GT. LIMMLT) NOMLT = LIMMLT                            PLOT  813
      IPTR = IOPTR                                                      MODA  593
      DO 90 I=1,NOMLT                                                   PLOT  814
      CALL DMGET(IWHOLE2,IPTR,NAME)                                     MODA  594
      CALL DMGET(IWHOLE1,IPTR,IPTR)                                     MODA  595
      ENCODE(25,75,IHOLD) I,NAME                                        PLOT  816
   75 FORMAT(1H0,I2,7H NAME= ,A10)                                      PLOT  817
      WRITE(6,40) (IHOLD(K),K=1,2)                                      PLOT  818
   90 CONTINUE                                                          PLOT  819
```

```
      95 RETURN                                                          PLOT      820
C                                                                        PLOT      821
66    C        ERROR CALLS                                               PLOT      822
 9991 CALL ERROR(37H(32H PARAMETER NO NOT FOUND IN TABLE),-3,5HNOPTS)    PLOT      823
      END                                                                PLOT      824
```

```
      SUBROUTINE PLOT3, RETURNS(FIN)                                    PLOT    825
C                                                                       PLOT    826
C        THIS SUBROUTINE UPDATES THE PLOT GROUP POINTER TO CONTAIN THE   PLOT    827
C        ADDRESS OF THE NEXT PG BEAD IN THE DATA STRUCTURE. IF NONE IS  PLOT    828
C        FOUND A RETURN IS MADE VIA FIN.                                PLOT    829
C                                                                       PLOT    830
      COMMON/HEADPTR/IFPG                                               PLOT    831
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT    832
      COMMON/STATSWT/ ISTAT,ISTATM,ISTATP,STRSWT                        PLOT    833
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                PLOT    834
      COMMON/CMBODS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,    PLOT    835
     X IWHOLE6                                                          PLOT    836
C                                                                       PLOT    837
C        CHECK IF FIRST INITIALIZATION                                  PLOT    838
      IF (.NOT. ISTAT) GO TO 10                                         PLOT    839
C                                                                       PLOT    840
C        INITIALIZE PG POINTER                                          PLOT    841
      IPGPT = IFPG                                                      PLOT    842
      GO TO 20                                                          PLOT    843
C                                                                       PLOT    844
C        NOT FIRST INITIALIZATION                                       PLOT    845
   10 ISTAT = .TRUE.                                                    PLOT    846
C                                                                       PLOT    847
C        ZERO OUT CVECTOR                                               PLOT    848
      ICONPT=IAXSPT= IPMPT = 0                                          PLOT    849
C                                                                       PLOT    850
C        GET NEXT PG BEAD                                               PLOT    851
      CALL CMGET(IWHOLE1,IPGPT,IPGPT)                                   PLOT    852
C                                                                       PLOT    853
C        END OF DATA STRUCTURE, RETURN FIN                              PLOT    854
      IF (IPGPT .EQ. 0) RETURN FIN                                      PLOT    855
C                                                                       PLOT    856
C        UPDATE POINTER TO CONDITION TEXT BEAD                          PLOT    857
   20 CALL CMGET(IWHOLE5,IPGPT,ICONPT)                                  PLOT    858
      RETURN                                                            PLOT    859
      END                                                               PLOT    860
```

```
      SUBROUTINE PLOTER                                                 PLOT      861
C                                                                       PLOT      862
C        THIS SUBROUTINE CONTAIN THE LOGICAL SEQUENCING OF THE PLOTTING PLOT      863
C        OPERATION AND INFORMATION DISPLAY.                             PLOT      864
C                                                                       PLOT      865
      COMMON/AXESIND/ NOFYAX,NOYAX,IX,IY(20)                            PLOT      866
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                         PLOT      867
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                PLOT      868
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPTR                    PLOT      869
      COMMON/PMTEXT/ LENGTH,MNAMES,MSTRUCT,MTEXT,NAMEP                  PLOT      870
      COMMON/STATS/ ITOTPTS,ITOTDEL,NTPBL,NUMINPG,ITOTPLT               PLOT      871
      COMMON/CMRCDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT      872
     X IWHOLE6                                                          PLOT      873
C                                                                       PLOT      874
C        INITIALIZE VARIABLES                                           PLOT      875
      LIMCOND = 11                                                      PLOT      877
      LIMPM   = 11                                                      PLOT      878
C                                                                       PLOT      885
C        CHECK WHETHER HEAD AND DATA POINTS EXISTS                      PLOT      886
      IF (IPTR .NE. 0) GO TO 30                                         PLOT      887
      I = 0                                                             PLOT      888
      GO TO 70                                                          PLOT      889
C                                                                       PLOT      894
C        SPECIFY ORIGIN                                                 PLOT      895
   30 XOIST = 12.0                                                      PLOT      896
      IF (NOYAX .GT. 3) XOIST = 11.5 + FLOAT(NOYAX) * .5                PLOT      897
      CALL PLOT(XOIST,0.0,-23)                                          PLOT      898
      CALL FRAME                                                        PLOT      899
      NTPBL = NTPBL +1                                                  PLOT      900
C                                                                       PLOT      901
C        DRAW X AND Y AXES                                              PLOT      902
      CALL DISAXES                                                      PLOT      903
C                                                                       PLOT      904
C        EJECT TO TOP OF OUTPUT PAGE                                    PLOT      905
      WRITE(6,35)                                                       PLOT      906
   35 FORMAT('1')                                                       PLOT      907
C                                                                       PLOT      908
C        GET FIRST CURVE HEAD                                           PLOT      909
      CALL CMGET(IWHOLE0+11,IPTR,ICBPTR)                                NODA      596
      ICPTR = ICBPTR                                                    PLOT      911
      I = 0                                                             PLOT      912
C                                                                       PLOT      913
C        PLOT POINTS                                                    PLOT      914
   45 CALL DISPTS(ICBPTR)                                               PLOT      915
      I = I+ 1                                                          PLOT      916
C                                                                       PLOT      917
C        GET NEXT CURVE HEAD                                            PLOT      918
      CALL CMGET(IWHOLE1,ICBPTR,ICBPTR)                                 PLOT      919
      IF (ICBPTR .NE. 0) GO TO 45                                       PLOT      920
C                                                                       PLOT      921
C        DETERMINE IF CONDS AND PMS FIT ON PLOT                         PLOT      922
      CALL CMGET(IWHOLE5,IPGPT,NOFCON)                                  PLOT      923
      CALL CMGET(IWHOLE0+8,IPGPT,NOFLBL)                                PLOT      924
      NOFLBL = 2 * NOFLBL                                               PLOT      925
      IF (LENGTH+I*NOFYAX+NOFLBL .GT. LIMPM .OR. NOFCON .GT. LIMCOND)   PLOT      926
     X I = -I                                                           PLOT      927
C                                                                       PLOT      928
```

```
C       LABEL PLOT & PRINT OUT STATISTICS                          PLOT      929
   70 CALL DISINFO(I,IOPTR)                                        PLOT      930
60        IF (I .EQ. 0) GO TO 80                                   PLOT      931
          NUMINPG = NUMINPG +1                                     PLOT      932
C                                                                  PLOT      933
C         UPDATE PLOT BEAD POINTER                                 PLOT      934
          CALL DMGET(IWHOLE1,IPTR,IPTR)                            PLOT      935
65        IF (IPTR .NE. 0) GO TO 30                                PLOT      936
C                                                                  PLOT      937
C         SET STATUS SWITCH TO INDICATE UPDATE NEEDED              PLOT      938
   80 ISTAT = .FALSE.                                              PLOT      939
          RETURN                                                   PLOT      940
70        END                                                      PLOT      941
```

```
      SUBROUTINE PM, RETURNS(NEXT)                                      PLOT    942
C                                                                       PLOT    943
C     THIS SUBROUTINE UPDATES THE PM POINTER TO CONTAIN THE ADDRESS OF  PLOT    944
C     THE NEXT PM BEAD FOR A GIVEN AXES BEAD. IF NONE IS FOUND A RETURN PLOT    945
C     NEXT IS TAKEN.                                                    PLOT    946
C                                                                       PLOT    947
      COMMON/STATSWT/ ISTAT,ISTATM,ISTATP,STRSWT                        PLOT    948
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                PLOT    949
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT    950
      COMMON/ CMPOOS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT    951
     X IWHOLE6                                                          PLOT    952
C                                                                       PLOT    953
C        INITIALIZATION OR UPDATE                                       PLOT    954
      IF (ISTAT) GO TO 30                                               PLOT    955
C                                                                       PLOT    959
C        UPDATE CURRENT PM POINTER                                      PLOT    960
      CALL BMGET(IWHOLE1,IPMPT,IPMPT)                                   PLOT    961
C                                                                       PLOT    962
C        CHECK IF PM BEADS ARE EXHAUSTED FOR CURRENT AXES BEAD          PLOT    963
      IF (IPMPT .EQ. 0) RETURN NEXT                                     PLOT    964
      ISTAT = .TRUE.                                                    PLOT    965
      GO TO 50                                                          PLOT    966
C                                                                       PLOT    967
C        INITIALIZE PM BEAD FROM NEW AXES BEAD                          PLOT    968
   30 CALL BMGET(IWHOLE4,IAXSPT,IPMPT)                                  PLOT    969
   50 CONTINUE                                                          PLOT    970
C                                                                       MODA    597
C        INITIALIZE P-M SWITCHES                                        MODA    598
      ISTATM = ISTATP = .FALSE.                                         MODA    599
      STRSWT = .FALSE.                                                  MODA    600
      RETURN                                                            PLOT    975
      END                                                               PLOT    976
```

```
      SUBROUTINE PRNLIST(LSTPTR,IND)                                    PLOT  1050
C                                                                       PLOT  1051
C        IND=0 - PRUNE LIST AND FN BEADS                                PLOT  1052
C        IND=1 - PRUNE FN BEADS ONLY                                    PLOT  1053
C        LSTPTR - POINTER TO BEGINNING OF CONNECTED 3 WORD LIST BEADS   PLOT  1054
C                                                                       PLOT  1055
      COMMON/CMFCDS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,   PLOT  1056
     X IWHOLE6                                                          PLOT  1057
C                                                                       PLOT  1058
C        GET FN POINTER                                                 PLOT  1059
   10 CALL DMSET(IWHOLE3,LSTPTR,IFNPTR)                                 PLOT  1060
      CALL DMSET(IWHOLE3,LSTPTR,0)                                      PLOT  1061
C                                                                       PLOT  1062
C        LOOP THROUGH FN LIST DELETING ALL BEADS                        PLOT  1063
   20 IF (IFNPTR .EQ. 0) GO TO 30                                       PLOT  1064
      CALL DMGET(IWHOLE1,IFNPTR,NFNPTR)                                 PLOT  1065
      CALL DMRL9D(IFNPTR)                                               PLOT  1066
      IFNPTR = NFNPTR                                                   PLOT  1067
      GO TO 20                                                          PLOT  1068
C                                                                       PLOT  1069
C        LOOP TO NEXT LIST BEAD DELETING CURRENT LIST BEAD IF REQUESTED PLOT  1070
   30 CALL DMGET(IWHOLE1,LSTPTR,NLSTPTR)                                PLOT  1071
      IF (IND .EQ. 0) CALL DMRL9D(LSTPTR)                               PLOT  1072
      LSTPTR = NLSTPTR                                                  PLOT  1073
      IF (LSTPTR .NE. 0) GO TO 10                                       PLOT  1074
      RETURN                                                            PLOT  1075
      END                                                               PLOT  107
```

```
1                SUBROUTINE PRNTPTS(XPTS,YPTS,RECNO,NOFPTS)                    PLOT    1077
                 DIMENSION XPTS(1),YPTS(1),RECNO(1)                            PLOT    1078
                 INTEGER RECNO                                                 PLOT    1079
                 DO 10 J=1,NOFPTS                                              PLOT    1080
5                WRITE(6,5) XPTS(J),YPTS(J),RECNO(J)                           PLOT    1081
               5 FORMAT(1H ,2(3X,G13.7),7X,I5)                                 PLOT    1082
              10 CONTINUE                                                      PLOT    1083
                 RETURN                                                        PLOT    1084
                 END                                                           PLOT    1085
```

```
      PROGRAM PS4MAIN                                                   PLOT    1086
C                                                                       PLOT    1087
C        THIS PROGRAM IS THE DRIVER ROUTINE FOR THE ACCUMULATION OF     PLOT    1088
C        STATISTICS, PREPARATION OF PLOTS, AND DISPLAY OF PLOTTING      PLOT    1089
C        INFORMATION THAT IS OUTPUTED BY THE ENTIRE SYSTEM PACKAGE      PLOT    1090
C                                                                       PLOT    1091
      COMMON/HEADPTR/IFPG                                               PLOT    1092
      COMMON /PLTSZ/ VFACTR                                             PLOT    1093
      COMMON/CVECTOR/ IPGPT,ICONPT,IAXSPT,IPMPT,IPLTPTR                 PLOT    1094
      COMMON/STATSWT/ISTAT,ISTATM,ISTATP,STRSWT                         PLOT    1095
      LOGICAL ISTAT,ISTATM,ISTATP,STRSWT                                PLOT    1096
      COMMON/PSTEXT/LENGTH,MNAMES,MSTRUCT,MTEXT, NAMEP                  PLOT    1097
      COMMON/STATS/ITOTPTS,ITOTDEL,NTPBL,NUMINPG,ITOTPLT                PLOT    1098
      COMMON/ITABLE/ IELETAB,IFUNTAB,XSCL,YSCL                          PLOT    1099
      COMMON/LIMIT/XMIN,XMAX,YMIN,YMAX                                  PLOT    1100
      COMMON/CHRCOS/IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,    PLOT    1101
     X IWHOLE6                                                          PLOT    1102
      COMMON /INITSW/ LISTSW,SWITCH,NOOFTP                              PLOT    1103
      COMMON/PTRSZ/ NPTING0                                             PLOT    1104
      LOGICAL SWITCH                                                    PLOT    1105
      DIMENSION DATO(1024)                                              PLOT    1106
      DATA IPGPT,ICONPT,IAXSPT,IPMPT/4*(0)/                             PLOT    1107
      DATA ISTAT,ISTATM,ISTATP/.TRUE.,.FALSE.,.FALSE./                  PLOT    1108
      DATA LENGTH,MNAMES,MSTRUCT,MTEXT,NAMEP/5*(0)/                     PLOT    1109
      DATA ITOTPTS,ITOTDEL,NTPBL,NUMINPG,ITOTPLT/5*(0)/                 PLOT    1110
      DATA IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,IWHOLE6      PLOT    1111
     X/0100000000008,0100000000019,0100000000028,0100000000038,        PLOT    1112
     X 0100000000048,0100000000058,0100000000068/                      PLOT    1113
C                                                                       MODA     601
C        SORT ELEMENT TABLE ON NAME                                     MODA     602
      CALL SORTAB(IELETAB,1)                                            MODA     603
C                                                                       PLOT    1114
C        INITIALIZE PLOT PACKAGE                                        PLOT    1115
         NTPBL = 3                                                      PLOT    1116
      CALL FACTOR(VFACTR)                                               PLOT    1117
      CALL SSWTCH(1,J)                                                  PLOT    1118
      IF (J .EQ. 2) CALL PLOTS(DATO,1024,7)                             PLOT    1119
      CALL PLOT(0.0,1.0,-23)                                            PLOT    1120
      ISTATM = ISTATP = .FALSE.                                         PLOT    1121
      GO TO 25                                                          PLOT    1122
C                                                                       PLOT    1123
C        UPDATE THE TOTAL NUMBER OF PLOTS GENERATED                     PLOT    1124
   15 ITOTPLT = ITOTPLT + NUMINPG                                       PLOT    1125
C                                                                       PLOT    1126
C        WRITE OUT THE NUMBER OF PLOTS IN THE PLOT GROUP                PLOT    1127
      CALL DISTOT1(NUMINPG)                                             PLOT    1128
C                                                                       PLOT    1129
C        UPDATE PLOT GROUP POINTER                                      PLOT    1130
   25 CALL PLOT3,RETURNS(99990)                                         PLOT    1131
C                                                                       PLOT    1132
C        UPDATE REMAINING DATA STRUCTURE POINTERS                       PLOT    1133
   30 CALL AXES, RETURNS(15)                                            PLOT    1134
   45 CALL PM, RETURNS(30)                                              PLOT    1135
   60 CALL PLOT3D, RETURNS(45)                                          MODA     604
C                                                                       PLOT    1137
C        PLOT AND DISPLAY INFORMATION FLAGGED BY POINTERS               PLOT    1138
      CALL PLOTER                                                       PLOT    1139
```

```
      GO TO 60                                                          PLOT    1140
C                                                                       PLOT    1141
60    C     CLOSE PLOT PACKAGE                                          PLOT    1142
99999 CALL SSWTCH(1,J)                                                  PLOT    1143
      IF (J .EQ. 1) CALL PLOT(0.0,0.0, -3)                              PLOT    1144
      IF (J .EQ. 2) CALL PLOTE(IJKLMN)                                  PLOT    1145
C                                                                       PLOT    1146
65    C     WRITE OUT THE TOTAL NUMBER OF PLOTS GENERATED               PLOT    1147
      CALL DISTOT2(ITOTPLT)                                             PLOT    1148
      END                                                               PLOT    1149
```

```
      SUBROUTINE RANGE(XPTS,YPTS,RECNO,NPTS)                                MOOB     40
C                                                                           PLOT   1151
C        THIS SUBROUTINE SEARCHES THROUGH AN ARRAY OF X AND Y POINTS        PLOT   1152
C        EXCLUDING ANY WHOSE VALUES LIE OUTSIDE OF PREDETERMINED LIMITS     PLOT   1153
C                                                                           PLOT   1154
C        XPTS - ARRAY OF X VALUES                                           PLOT   1155
C        YPTS - ARRAY OF ASSOCIATED Y VALUES                                PLOT   1156
C        RECNO - ARRAY OF ASSOCIATED RECORD NUMBERS                         MOOB     41
C        NPTS - NUMBER OF POINTS IN EACH ARRAY, PASSED IN                   PLOT   1157
C             - NUMBER OF POINTS DELETED FROM EACH ARRAY, PASSED OUT        PLOT   1158
C                                                                           PLOT   1159
      INTEGER RECNO                                                         MOOB     42
      DIMENSION XPTS(NPTS),YPTS(NPTS),RECNO(NPTS)                           MOOB     43
      DIMENSION ENDPT(4)                                                    MOOB     44
      COMMON /CMPOOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,      MOOB     45
     X IWHOLE6                                                              MOOB     46
      COMMON /CVECTOR/ IPGPT,ICONPT,IAXSPT,IPNPT,IPTR                       MOOB     47
      COMMON /AXESIND/ NOFYAX,NO,IX,IY(20)                                  MOOB     48
      COMMON/LIMIT/XMIN,XMAX,YMIN,YMAX                                      PLOT   1161
C                                                                           MOOB     49
C        DETERMINE IF AXES SCALE READ CAN CONTAIN CROSS PLOT INFO           MOOB     50
      IF (NOFYAX .NE. 1) GO TO 15                                           MOOB     51
      CALL DMGET(IWHOLE0+9,IPTR,IAXSL)                                      MOOB     52
      XMINO = XMAX                                                          MOOB     53
      XMAXO = XMIN                                                          MOOB     54
      YMINO = YMAX                                                          MOOB     55
      YMAXO = YMIN                                                          MOOB     56
      SUM = 0.                                                              MOOB     57
      DO 5 I=5,8                                                            MOOB     58
      CALL DMGET(IWHOLE0+I,IAXSL,ENDPT(I-4))                                MOOB     59
      SUM = SUM + ABS(ENDPT(I-4))                                           MOOB     60
    5 CONTINUE                                                              MOOB     61
      IF (SUM .NE. 0.) GO TO 15                                             MOOB     62
      ENDPT(1) = XMINO                                                      MOOB     63
      ENDPT(2) = XMAXO                                                      MOOB     64
      ENDPT(3) = YMINO                                                      MOOB     65
      ENDPT(4) = YMAXO                                                      MOOB     66
   15 CONTINUE                                                              MOOB     67
C        SEARCH THROUGH X-Y ARRAYS DELETING OUT OF LIMIT POINTS             PLOT   1162
C                                                                           PLOT   1163
      IPLACE = 1                                                            PLOT   1164
      DO 40 ILOOK =1,NPTS                                                   PLOT   1165
C                                                                           PLOT   1166
C        CHECK FOR POINTS OUT OF BOUNDS                                     PLOT   1167
      IF (XPTS(ILOOK) .LT. XMIN .OR. XPTS(ILOOK) .GT. XMAX) GO TO 40        PLOT   1168
      IF (YPTS(ILOOK) .LT. YMIN .OR. YPTS(ILOOK) .GT. YMAX) GO TO 40        PLOT   1169
C                                                                           PLOT   1170
C        POINTS IN RANGE                                                    PLOT   1171
      IF (ILOOK .EQ. IPLACE) GO TO 30                                       PLOT   1172
   20 XPTS(IPLACE) = XPTS(ILOOK)                                            PLOT   1173
      YPTS(IPLACE) = YPTS(ILOOK)                                            PLOT   1174
      RECNO(IPLACE) = RECNO(ILOOK)                                          MOOB     68
C                                                                           MOOB     69
C        UPDATE MAX AND MIN DATA END POINTS                                 MOOB     70
   30 IF (NOFYAX .NE.1) GO TO 35                                            MOOB     71
      IF (XPTS(IPLACE) .LT. ENDPT(1)) ENDPT(1) = XPTS(IPLACE)               MOOB     72
      IF (XPTS(IPLACE) .GT. ENDPT(2)) ENDPT(2) = XPTS(IPLACE)               MOOB     73
```

```
                  IF (YPTS(IPLACE) .LT. ENDPT(3)) ENDPT(3) = YPTS(IPLACE)          MODB     74
                  IF (YPTS(IPLACE) .GT. ENDPT(4)) ENDPT(4) = YPTS(IPLACE)          MODB     75
60                35 IPLACE = IPLACE + 1                                           MODB     76
                  40 CONTINUE                                                      PLOT   1176
                C                                                                  MODB     77
                C     ZERO OUT UNUSED LOCATIONS                                    MODB     78
                  IF (IPLACE-1 .EQ. NPTS) GO TO 60                                 MODB     79
65                DO 50 I=IPLACE,NPTS                                              MODB     80
                  XPTS(I) = 0.                                                     MODB     81
                  YPTS(I) = 0.                                                     MODB     82
                  RECNO(I) = 0                                                     MODB     83
                  50 CONTINUE                                                      MODB     84
70                60 NPTS = NPTS - IPLACE + 1                                      MODB     85
                C                                                                  MODB     86
                C     UPDATE SNCPTS OF DATA IN SCALE BEAD                          MODB     87
                  IF (NOFYAX .NE.1) GO TO 80                                       MODB     88
                  DO 70 I=5,9                                                      MODB     89
75                CALL ONSET(IWHOLE0+I,IAXSL,ENDPT(I-4))                           MODB     90
                  70 CONTINUE                                                      MODB     91
                  80 CONTINUE                                                      MODB     92
                  RETURN                                                           PLOT   1180
                  END                                                              PLOT   1181
```

```
      SUBROUTINE TEXT(ICHAR,LEVCNT,INAME,VAL,IHOLD)                          MODB       93
      DIMENSION IHOLD(3)                                                     MODA      606
      COMMON /COMPCOS/ IWHOLE0,IWHOLE1,IWHOLE2,IWHOLE3,IWHOLE4,IWHOLE5,      MODB       94
     X IWHOLE6                                                               MODB       95
      DATA ID /4HTEXT/                                                       MODA      607
C                                                                            MODA      608
C        GET MULT ELEMENT NAME                                               MODA      609
      CALL DMGET(IWHOLE0+11+2*LEVCNT,IPLTPTR,INAME)                          MODA      610
C                                                                            MODA      611
C        GET MULT VALUE                                                      MODA      612
      CALL DMGET(IWHOLE0+12+2*LEVCNT, IPLTPTR, VAL)                          MODA      613
C                                                                            MODA      614
C        CHECK FOR OFF SURFACE SETTING                                       MODA      615
      IF (VAL .NE. 999.) GO TO 60                                            MODA      616
      IWORD = INAME                                                          MODA      617
      CALL SEARCH(IELETAB, 1, IWORD)                                         MODA      618
      IF (IWORD .EQ. 0) GO TO 9991                                           MODA      619
      IELEN = LOADVAL(IELETAB, 2, IWORD)                                     MODA      620
      IF (IELEN .LE. 600) GO TO 61                                           MODB       96
C                                                                            MODA      622
C        ENCODE BCD TEXT INTO HOLD ARRAY - OFF VALUE                         MODA      623
      ENCODE (30,50,IHOLD) ICHAR,LEVCNT,INAME                                MODB       97
   50 FORMAT(A1,I2,1X,A10,5H= OFF,11X)                                       MODB       98
      GO TO 80                                                               MODA      626
C                                                                            MODA      627
C        ENCODE BCD TEXT INTO HOLD ARRAY                                     MODA      628
   60 ENCODE(30,70,IHOLD) ICHAR,LEVCNT,INAME,VAL                             MODB       99
   70 FORMAT(A1,I2,A10,2H= ,G12.6,2X)                                        MODB      [illegible]
   80 CONTINUE                                                               MODA      [illegible]
      RETURN                                                                 MODA      632
C                                                                            MODA      633
C        ERROR CALLS                                                         MODA      634
 9991 CALL ERROR(31H(26H ELEMENT NOT IN NAME TABLE),-3,ID)                   MODA      635
      END                                                                    MODA      636
```

APPENDIX B

PLUGO PHASE II LISTING

```
      OVERLAY(OVERLAY,0,0)                                                    STARTER     2
      PROGRAM STARTER (OUTPUT,TAPE6=64,TAPE99,TAPE43=64)                      23FEB       2
      COMMON/DMTBL/FILES(8)                                                   NSROC       1
      DIMENSION DMBLK (4265)                                                  NSROC       2
      COMMON/SLFIT/ICR12,ICHNG                                                /SLFIT/     2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                      /ITRIMT/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG                30AUG       1
C     J=I**2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/     3
      COMMON/TDIS/ITRIM1,PRNAME,NX,NY,XMN,DELX,YMN,DELY,AAA(4)                1SEPT       3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1          10MAY77    61
     1,IDELTE,YDELBC(2)                                                       FLCAD2      5
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)       /CROSTB/    2
     1,ERACH,PVALU(10),PNAME(10),IRD(10)                                      /CROSTB/    3
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1,      /AXBEAD/    2
     X              IAXY2,IAXY3                                               /AXBEAD/    3
      COMMON/AXSBEAD/IAXSXM,IAXSXI,IAXSY1M,IAXSY1I,IAXSY2M,IAXSY2I,           /AXSBEAD/   2
     X               IAXSY3M,IAXSY3I                                          /AXSBEAD/   3
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                          /AXSCALE/   2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR       /BEADS/     2
      COMMON/BLOCK/IBLOCK(50)                                                 /BLOCK/     2
      COMMON/COMPCOD/IWDZERO,IWDONE,IWDTWO,IWDTHRE,IWDFOUR,IWDFIVE,           /COMPCOD/   2
     X               IWDSIX,IWDSEVE,IWDEIGH,IWDNINE,IWDTEN                    /COMPCOD/   3
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,        /CRVBEAD/   2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                       /CRVBEAD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(99),NN,NNEND(10),IDSPEC(19),IDCF(41),       /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),                6JULY77     2
     Y               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM       1SEPT       1
      COMMON/FITBEAD/IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA                     14APR76     1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                        CY7777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                       6JULY77     3
      COMMON/OPTION/ICPTION,ISHINE                                            LASTBUX     1
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                      /PARMSV/    2
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,          /PGBEAD/    2
     X              IPGLCA                                                    /PGBEAD/    3
      COMMON /PLOTCOM/ IFIRST,IPLTFL1,NCPC,ICALBD,OPLOT(128)                  /PLOTCOM/   2
      COMMON/PVAL/PPVALU(10),IPPV,PXNAM,NCRVS2                                16NOV       2
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,            /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,             /PLTBEAD/   3
     X               IPFMN,IPFMV                                              /PLTBEAD/   4
      COMMON/PNVAL/PPNAM,PPVAL                                                2NOV        4
      COMMON/PMBEAD/IPMNEXT,IPMPREV,IPMPLST,IPMMLST,IPMPLOT,IPMPTXT,          /PMBEAD/    2
     X              IPMMTXT                                                   /PMBEAD/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                   /POINTS/    2
      COMMON /SYMTAB/ ISYMTAB(20)                                             /SYMTAB/    2
      COMMON/MULTKS/NWS,AXK(21)                                               27MAY       1
      COMMON /SAVCPPT/ NEXCPPT                                                /SAVCPPT/   2
C                                                                             STARTER    23
C         EXECUTE SUBROUTINE PRELIM TO ATTACH THE GRAPHICS CONSOLE TO         STARTER    24
C         THE JOB, INITIALIZE THE GRAPHICS PARAMETERS, INITIALIZE THE         STARTER    25
C         DATA MANAGER, AND INITIALIZE THE COMPONENT CODES FOR THE            STARTER    26
C         VARIOUS BEADS.                                                      STARTER    27
C                                                                             STARTER    28
      CALL AEXEC                                                              21JUNE76    2
      CALL DMINIT (6LDMFILE,DMBLK(1),4265,6)                                  NSROC       3
      NCEC=0                                                                  14MARCH     2
      IDCVR=0                                                                 20SEPT     24
      CALL PARV                                                               20SEPT     25
```

```
                IF (IRCVR.EQ.0) GO TO 10                                  20SEPT    26
              C                                                           25JULY77   2
   60         C     THIS SETS A FLAG TO ZERO SO THE MENU IN QTRBACK        25JULY77   3
              C     WILL BE DRAWN IN CASE OF A RECOVERY                   25JULY77   4
              C                                                           25JULY77   5
                IDMENU(1)=0                                               25JULY77   6
                ICR12=3                                                   SLOPES     2
   65           ICHNG=0                                                   SLOPES     3
                CALL RECPLUG                                              20SEPT    27
             10 IRCVR=1                                                   20SEPT    28
                CALL PRELIMS                                              STARTER   29
              C                                                           STARTER   30
   70           CALL AETSKC(5HINTRO)                                      STARTER   31
              C                                                           STARTER   32
              C       DISPLAY THE STOP BUTTON                             STARTER   33
              C                                                           STARTER   34
                CALL ORIGIN(0.,0.)                                        STARTER   35
   75           CALL ANNOTA(-.24,9.5,4HSTOP,4,ISTOP,IBUT,254,-0,-0)       14APR76    8
              C                                                           STARTER   37
              C       GET THE POINTER TO THE FIRST PLOT GROUP BEAD        STARTER   38
              C       PASS IT THROUGH COMMON/BEADS/ TO TASK PGSPECS.      STARTER   39
              C                                                           STARTER   40
   80           CALL DMGET (100000000013,100008,IPGPTR)                   NSRDC      4
                CALL AETSKC(7HPGSPECS)                                    STARTER   42
                PRINT 100                                                 27MAY76    1
             100 FORMAT(* X*)                                             27MAY76    2
                BACKSPACE 99                                              16NOV      3
   85         C ** FORCES LOADING OF ROUTINE PLOT IN THE ZERO OVERLAY      25JULY77   8
                CALL PLOT (0.,0.,1)                                       25JULY77   9
                END                                                       STARTER   43
```

```
1                SUBROUTINE RECPLUG                                    RECPLUG    2
                 RETURN                                                RECPLUG    3
                 END                                                   RECPLUG    4
```

```
      SUBROUTINE HOLD(MODERR,IEXCH)                                     HOLD       2
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *  HOLD       3
C                                                                       HOLD       4
C     THE PURPOSE OF THIS ROUTINE IS TO PRINT ERROR RECOVERY INFO       HOLD       5
C                                                                       HOLD       6
C                                                                       HOLD       7
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *  HOLD       8
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON /HOLDI/ IXMES,IYMES                                        HOLD      10
      DIMENSION IL(3),IOUT(5),IL2(4)                                    HOLD      11
      DIMENSION IEXCH(17)                                               HOLD      12
      INTEGER SHIFT                                                     HOLD      13
      DATA IL/10H    **** ER,10HROR RECOVE,10HRY BEGUN   /              HOLD      14
      DATA IL2/10H    **** EN,10HTER GO,STO,10HP,OMPX,OMP,              HOLD      15
     *  10H          /                                                  HOLD      16
      DATA MSKP/77777700000000000003/                                   HOLD      17
        DATA IFX/-600/,IFY/1200/                                        HOLD      18
        IXMES=IFX $ IYMES=IFY $ NCON=1                                  HOLD      19
    1 CALL GICNUM(NCON)                                                 HOLD      20
        CALL GIMASK(NCON,0,1,1)                                         HOLD      21
        CALL GIMASK(NCON,0,2,18)                                        HOLD      22
        CALL GIMASK(NCON,0,4,20)                                        HOLD      23
        CALL GIMASK(NCON,0,8,24)                                        HOLD      24
        CALL GIMASK(NCON,0,16,16)                                       HOLD      25
      CALL PUTLIN(3,IL)                                                 HOLD      26
C                                                                       HOLD      27
C******GET SYSTEM ERROR CODE                                           HOLD      28
C                                                                       HOLD      29
      IEL = SHIFT( IEXCH(3) .AND. MSKP,24 )                             HOLD      30
      IWORD=IEXCH(1) .AND. 00000 00000 00007 777778                     HOLD      31
      IF(IWORD .EQ. 2) GO TO 16                                         HOLD      32
      IADDR=SHIFT(IEXCH(1) .AND. 00777 77700 00000 000008,24)           HOLD      33
      IF(IWORD .EQ. 1) GO TO 14                                         HOLD      34
      ENCODE(30,12,IOUT) IWORD,IADDR                                    HOLD      35
   12 FORMAT(* *,7X,*SYSTEM ERROR CODE=*,02,* AT *,06)                  HOLD      36
   13 CALL PUTLIN(4,IOUT)                                               HOLD      37
      GO TO 30                                                          HOLD      38
   14 ENCODE(40,15,IOUT) IADDR                                          HOLD      39
   15 FORMAT(8X,*CP TIME LIMIT  AT  *,06,7X)                            HOLD      40
      GO TO 13                                                          HOLD      41
C                                                                       HOLD      42
C******GET TYPE AND ADDRESS AND RIGHT JUSTIFY                          HOLD      43
C                                                                       HOLD      44
   16 ITYPE=SHIFT(MODERR .AND. 77770 00100 00000 000008,12)             HOLD      45
      IADDR=SHIFT(MODERR .AND. 00007 77777 00000 000008,30)             HOLD      46
C                                                                       HOLD      47
C******NOW NOTIFY USER OF HIS PROBLEM                                  HOLD      48
C                                                                       HOLD      49
      ENCODE(42,20,IOUT) ITYPE,IADDR                                    HOLD      50
   20 FORMAT(* *,7X,*ARITHMETIC MODE ERROR=*,02,* AT *,06)              HOLD      51
      CALL PUTLIN(5,IOUT)                                               HOLD      52
   30 CALL PUTLIN(4,IL2)                                                HOLD      53
      CALL GETLIN(1,IANS)                                               HOLD      54
      IF(IANS .EQ. 2HGO .OR. IANS .EQ. 1HG) RETURN                      HOLD      55
      IF (IANS.EQ.4HSTOP.OR.IANS.EQ.1HS)  CALL STOP                     HOLD      56
      IF(IANS .EQ. 4HDMPX) CALL D4PX(IEXCH)                             HOLD      57
```

```
                IF(IABS .EQ. 3HOMP) CALL OMP(MODERR,IFL)                HOLD      58
                GO TO 30                                                HOLD      59
60              END                                                     HOLD      60
```

```
      SUBROUTINE PUTLIN(N,IOUT)                                          PUTLIN      2
      DIMENSION IOUT(1)                                                  PUTLIN      3
      COMMON /HOLDI/ IXMES,IYMES                                         PUTLIN      4
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CY8777      1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77     3
      NCON=1 ! NBYTE=0 $ MBYTE=1500                                      PUTLIN      6
      CALL GURSET(IXMES,IYMES,28,IBUF,NBYTE,MBYTE)                       PUTLIN      7
      NCHAR=10*N                                                         PUTLIN      8
      CALL GUAN(IOUT,NCHAR,IBUF,NBYTE,MBYTE)                             PUTLIN      9
      CALL GIDISP(NCON,IBUF,NBYTE,IDD,1,99,-0,-0)                        PUTLIN     10
      IYMES=IYMES-50                                                     PUTLIN     11
       IF(IYMES.GT.-1200) RETURN                                         PUTLIN     12
    1 CALL GICNJB(NCON)                                                  PUTLIN     13
      CALL GIMASK(NCON,0,1,1)                                            PUTLIN     14
      CALL GIMASK(NCON,0,2,18)                                           PUTLIN     15
      CALL GIMASK(NCON,0,4,20)                                           PUTLIN     16
      CALL GIMASK(NCON,0,8,24)                                           PUTLIN     17
      CALL GIMASK(NCON,0,16,16)                                          PUTLIN     18
      IYMES=1200                                                         PUTLIN     19
       RETURN                                                            PUTLIN     20
      ENTRY GETLIN                                                       PUTLIN     21
      NCON=1 ! NCHAR=10*N                                                PUTLIN     22
      CALL GIEOM (NCON,IEOM,8,200)                                       CY8777      3
  299 CALL GIANS(NCON,NCHAR,IXMES,IYMES)                                 PUTLIN     24
  300 CALL GIBUT(0,NCON,IODT,IODC)                                       PUTLIN     25
      IF(IODC.NE.200) GO TO 300                                          PUTLIN     26
      CALL GIANE(NCON,NCHAR,IOUT)                                        PUTLIN     27
      NB=NCHAR                                                           CY8777      4
      NW=NB/10                                                           PUTLIN     29
      J=MOD(NB,10)                                                       PUTLIN     30
      IF(J.EQ.0) RETURN                                                  PUTLIN     31
      NW=NW+1                                                            PUTLIN     32
      ENCODE(10,111,FMT) J                                               PUTLIN     33
  111 FORMAT(2H(A,I1,1H))                                                PUTLIN     34
      DECODE(10,FMT,IOUT(NW)) IOUT(NW)                                   PUTLIN     35
      RETURN                                                             PUTLIN     36
      END                                                                PUTLIN     37
```

```
1                BLOCK DATA                                                    BLOCKD     2
                 COMMON /SYMTAB/ ISYMTAB(20)                                   /SYMTAB/   2
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG      3CAUG      1
           C     J=N*2**18  OMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/    3
5                DATA IBD1/0/                                                  3NOV       2
                 DATA ISYMTAB/1H+,1HX,1H0,1H*,1HS,1HZ,1HH,1HC,1HR,1HD,1HQ,     BLOCKD     4
                1    1HA,1HB,1HS,1HP,1H8,1H3,1HU,1HM,1HY/                      BLOCKD     5
                 END                                                           BLOCKD     6
```

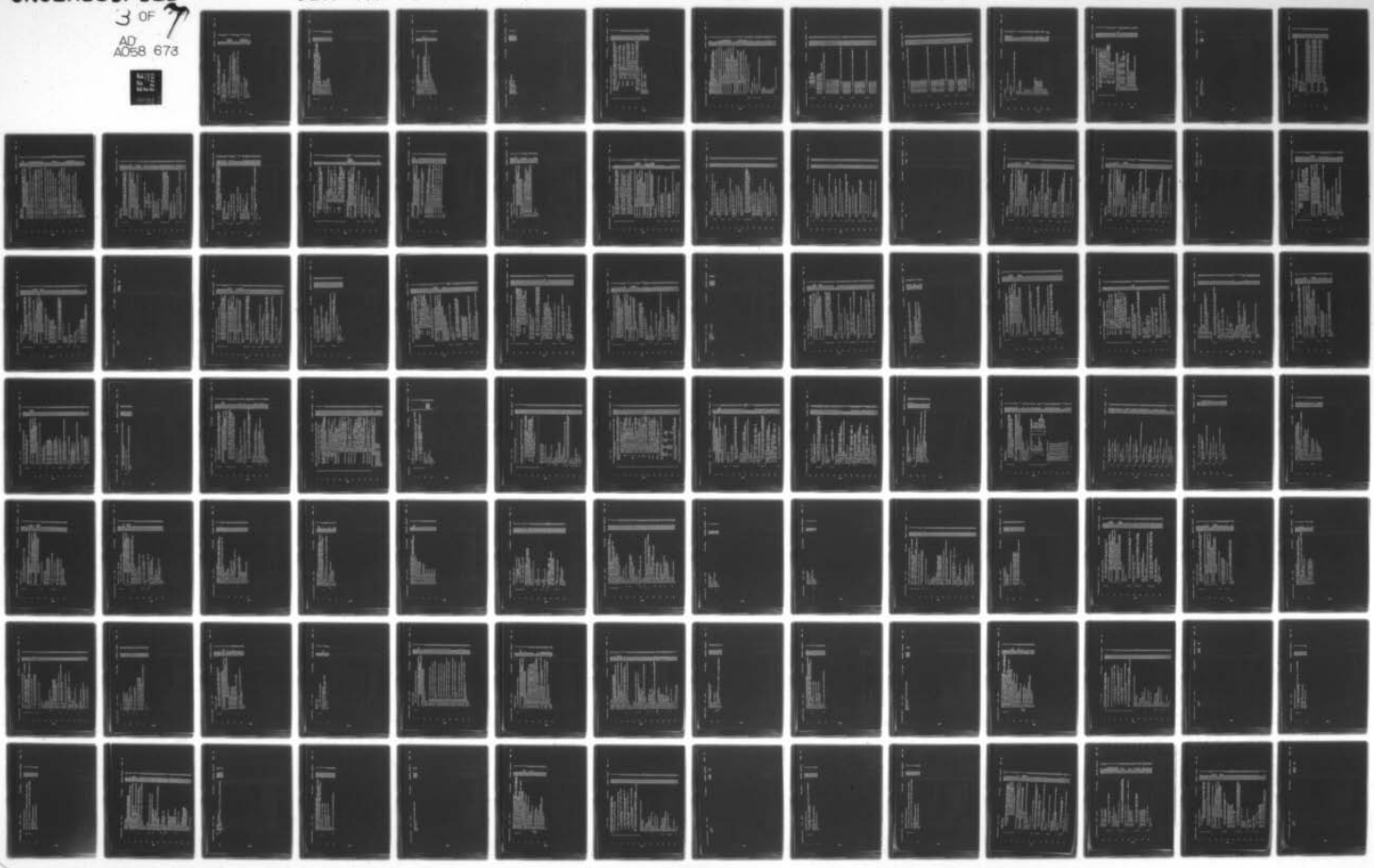
```
      SUBROUTINE ANNOTA(XINCH,YINCH,STRING,NCHAR,IDDAD,IDDT,                ANNOTA     2
     X                  IDDC,IDWA,IDWB)                                    ANNOTA     3
C                                                                           ANNOTA     4
C     ANNOTA DISPLAYS A STRING OF ALPHANUMERIC CHARACTERS ON THE            ANNOTA     5
C     SCREEN. EACH LINE REQUIRES A SEPARATE CALL TO ANNOTA.                 ANNOTA     6
C          XINCH - I - THE X-COORDINATE IN INCHES OF THE LOWER LEFT         ANNOTA     7
C                  CORNER OF THE FIRST CHARACTER IN THE STRING.             ANNOTA     8
C          YINCH - I - THE Y-COORDINATE IN INCHES OF THE LOWER LEFT         ANNOTA     9
C                  CORNER OF THE FIRST CHARACTER IN THE STRING.             ANNOTA    10
C          NCHAR - I - THE NUMBER OF CHARACTERS IN THE STRING.              ANNOTA    11
C                  THE MAXIMUM VALUE OF NCHAR IS 255.                       ANNOTA    12
C          IDDAD - O - THE SYSTEM DEFINED ASSOCIATIVE ADDRESS OF            ANNOTA    13
C                  THE DISPLAY ITEM.                                        ANNOTA    14
C           IDDT - I - THE DISPLAY TYPE.                                    ANNOTA    15
C           IDDC - I - THE DISPLAY CODE.                                    ANNOTA    16
C           IDWA - I - DISPLAY INFORMATION WORD A.                          ANNOTA    17
C           IDWB - I - DISPLAY INFORMATION WORD B.                          ANNOTA    18
C                                                                           ANNOTA    19
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                      CY6777     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                     6JULY77     3
      COMMON/PARMSV/XORUGU,YORUGU,UGUPIN                                    /PARMSV/   2
      INTEGER STRING(26)                                                    ANNOTA    22
      NBYTE = 0                                                             ANNOTA    23
      IXUGU = IFIX(XINCH * UGUPIN + XORUGU + .5)                            ANNOTA    24
      IYUGU = IFIX(YINCH * UGUPIN + YORUGU + .5)                            ANNOTA    25
      CALL GURSET(IXUGU,IYUGU,ICODE,IBUF,NBYTE,MBYTE)                       ANNOTA    26
      CALL GUAN(STRING,NCHAR,IBUF,NBYTE,MBYTE)                              ANNOTA
      CALL GIDISP(NCON,IBUF,NBYTE,IDDAD,IDDT,IDDC,IDWA,IDWB)                ANNOTA
      RETURN                                                                ANNOTA    29
      END                                                                   ANNOTA    30
```

```
      SUBROUTINE ERRORM(XINCH,YINCH,MESSAGE,NCHAR,ROUTINE,TYPE)          ERRORM   2
C                                                                        ERRORM   3
C         THIS ROUTINE DISPLAYS THE TRANSMITTED ERROR MESSAGE. IF THE    ERRORM   4
C         ERROR IS FATAL, PROCESSING IS TERMINATED AND THE ERROR MESSAGE ERRORM   5
C         IS DISPLAYED IN THE DAYFILE. IF THE ERROR IS INFORMATIVE, THEN ERRORM   6
C         PROCESSING CAN CONTINUE OR TERMINATE AT THE DISCRETION OF THE  ERRORM   7
C         USER.                                                          ERRORM   8
C             XINCH - I - THE X COORDINATE IN INCHES OF THE POINT ON THE ERRORM   9
C                     SCREEN WHERE THE LOWER LEFT CORNER OF AN           ERRORM  10
C                     INFORMATIVE ERROR MESSAGE WILL BE DISPLAYED.       ERRORM  11
C             YINCH - I - THE Y COORDINATE IN INCHES OF THE POINT        ERRORM  12
C                     DESCRIBED BY XINCH.                                ERRORM  13
C           MESSAGE - I - THE ERROR MESSAGE.                             ERRORM  14
C             NCHAR - I - THE NUMBER OF CHARACTERS IN THE ERROR MESSAGE. ERRORM  15
C                     THE LENGTH OF THE ERROR MESSAGE IS LIMITED TO 72   ERRORM  16
C                     CHARACTERS.                                        ERRORM  17
C           ROUTINE - I - THE NAME OF THE ROUTINE FROM WHICH ERRORM WAS  ERRORM  18
C                     CALLED.                                            ERRORM  19
C              TYPE - I - THE TYPE OF ERROR---5HFATAL FOR A FATAL ERROR  ERRORM  20
C                     OR 6HINFORM FOR AN INFORMATIVE ERROR.              ERRORM  21
C                                                                        ERRORM  22
      COMMON/BLOCK/IBLOCK(50)                                            /BLOCK/  2
      COMMON /GP/ IBUF(128),NCON,N8YTE,M8YTE,IGN,ISIN,                   CYB777   1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77  3
      INTEGER ROUTINE,TYPE,STRING(9),IDERROR(4),MESSAGE(6)               2NOV     6
      EQUIVALENCE (STRING,IBLOCK), (IDERROR,IBLOCK(9))                   ERRORM  26
      IF(NCHAR .GT. 72) NCHAR = 72                                       ERRORM  27
C                                                                        ERRORM  28
C         IS THE ERROR FATAL OR INFORMATIVE?                             ERRORM  29
C                                                                        ERRORM  30
      IF(TYPE .EQ. 5HFATAL) GO TO 30                                     ERRORM  31
C                                                                        ERRORM  32
C         THE ERROR IS INFORMATIVE. ENCODE                               ERRORM  33
C         THE ERROR MESSAGE AND DISPLAY IT.                              ERRORM  34
C                                                                        ERRORM  35
      ENCODE(80,1000,STRING) MESSAGE                                     ERRORM  36
 1000 FORMAT("ERROR---",7A10,A2)                                         ERRORM  37
      NCHAR = NCHAR + 8                                                  ERRORM  38
      ICODE = 1000                                                       ERRORM  39
      CALL ANNOTA(XINCH,YINCH,STRING,NCHAR,IDERROR(1),IGN,-0,-0,-0)      ERRORM  40
      CALL ANNOTA(XINCH,YINCH - .25,47HPICK EXECUTE TO CONTINUE. OTHERWI ERRORM  41
     XSE, PICK STOP.,47,IDERROR(2),IGN,-0,-0,-0)                         ERRORM  42
      CALL ANNOTA(XINCH + 2.,YINCH - .75,7HEXECUTE,7,IDERROR(3),IBUT,    ERRORM  43
     X            253,-0,-0)                                            14APR76  9
      ICODE = 1029                                                       ERRORM  45
C                                                                        ERRORM  46
C         WAIT FOR A BUTTON PICK FROM THE FOLLOWING MENU.                ERRORM  47
C              253 - EXECUTE --- CONTINUE PROCESSING                     2NOV     7
C              254 - STOP                                                2NOV     8
C                                                                        ERRORM  50
   10 CALL GINPUT(0,NCON,IDOT,IDOC)                                      ERRORM  51
      IF(IDOC .EQ. 253) GO TO 20                                         14APR76 10
      IF(IDOC .EQ. 254) CALL STOP                                        14APR76 11
      GO TO 10                                                           ERRORM  54
   20 CALL TERMSE(IDERROR,3)                                             ERRORM  55
      RETURN                                                             ERRORM  56
C                                                                        ERRORM  57
```

```
      C         THE ERROR IS FATAL.  PUT MESSAGE ON THE SCREEN AND          2NOV       9
      C         FORCE ERROR TO RETURN VIA RECOVERY PACKAGE.                 2NOV      10
60    C                                                                     2NOV      11
   30 CALL GICNJB (NCON)                                                    2NOV      12
      ENCODE (30,2000,STRING) ROUTINE                                       2NOV      13
 2000 FORMAT("FATAL ERROR****ROUTINE ",A7)                                  2NOV      14
      CALL ORIGIN (0.,0.)                                                   2NOV      15
65    C                                                                     14MARCH    3
      C       DISPLAY STOP BUTTON                                           14MARCH    4
      CALL ANNOTA (-.24,9.5,4HSTOP,4,IDSTOP,IBUT,254,-0,-0)                 14MARCH    5
      C                                                                     14MARCH    6
      CALL ANNOTA (-1.75,0.,STRING,30,IDERROR(1),IGN,-0,-0,-0)              2NOV      16
70    CALL ANNOTA (-5.,-1.,MESSAGE,NCHAR,IDERROR(2),IGN,-0,-0,-0)           2NOV      17
      CALL ANNOTA (-1.5,-2.,25HPICK EXECUTE TO CONTINUE.,                   2NOV      18
     1    25,IDERROR(3),IGN,-0,-0,-0)                                      2NOV      19
      CALL ANNOTA (-.5,-.3,7HEXECUTE,7,IDERROR(4),IBUT,253,-0,-0)           2NOV      20
   40 CALL GIBUT (0.,NCON,IDDT,IDDC)                                        2NOV      21
75    C                                                                     14MARCH    7
      C       RETURN TO PGSPECS                                             14MARCH    8
      C                                                                     14MARCH    9
      IF (IDDC.EQ.254) CALL STOP                                            14MARCH   10
      CALL TERASE (IDERROR,4)                                               14MARCH   11
80    CALL RETSKC (7HPGSPECS)                                               14MARCH   12
      RETURN                                                                2NOV      31
      END                                                                   ERRORM    73
```

AD-A058 673 DAYTON UNIV OHIO RESEARCH INST F/G 9/2
PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). VOLUME II--ETC(U)
MAY 78 J G JENSEN F33615-76-C-3076
UNCLASSIFIED UDR-TR-78-34 AFFDL-TR-78-55-VOL-2 NL

3 OF 7
AD A058 673



```
      C         THE ERROR IS FATAL.  PUT MESSAGE ON THE SCREEN AND          2NOV       9
      C         FORCE ERROR TO RETURN VIA RECOVERY PACKAGE.                 2NOV      10
60    C                                                                     2NOV      11
   30 CALL GICNJB (NCON)                                                    2NOV      12
      ENCODE (30,2000,STRING) ROUTINE                                       2NOV      13
 2000 FORMAT("FATAL ERROR****ROUTINE ",A7)                                  2NOV      14
      CALL ORIGIN (0.,0.)                                                   2NOV      15
65    C                                                                     14MARCH    3
      C       DISPLAY STOP BUTTON                                           14MARCH    4
      CALL ANNOTA (-.24,9.5,4HSTOP,4,IDSTOP,IBUT,254,-0,-0)                 14MARCH    5
      C                                                                     14MARCH    6
      CALL ANNOTA (-1.75,0.,STRING,30,IDERROR(1),IGN,-0,-0,-0)              2NOV      16
70    CALL ANNOTA (-5.,-1.,MESSAGE,NCHAR,IDERROR(2),IGN,-0,-0,-0)           2NOV      17
      CALL ANNOTA (-1.5,-2.,25HPICK EXECUTE TO CONTINUE.,                   2NOV      18
     1    25,IDERROR(3),IGN,-0,-0,-0)                                      2NOV      19
      CALL ANNOTA (-.5,-.3,7HEXECUTE,7,IDERROR(4),IBUT,253,-0,-0)           2NOV      20
   40 CALL GIBUT (0.,NCON,IDDT,IDDC)                                        2NOV      21
75    C                                                                     14MARCH    7
      C       RETURN TO PGSPECS                                             14MARCH    8
      C                                                                     14MARCH    9
      IF (IDDC.EQ.254) CALL STOP                                            14MARCH   10
      CALL TERASE (IDERROR,4)                                               14MARCH   11
80    CALL AETSKC (7HPGSPECS)                                               14MARCH   12
      RETURN                                                                2NOV      31
      END                                                                   ERRORM    73
```

```
1                SUBROUTINE G078G                                                G078G      2
          C      FOR INFORMATION CONTACT THOMAS S. ROWLAND  ACOS/56210           G078G      3
          C      TO MAKE ERROR 78 NON-FATAL,CALL G078G IN MAIN OVERLAY AFTER CALL T G078G    4
          C      THE CALL TO INITIAL DOES THE INITIALIZATION OF G078G AMONG OTHER T G078G    5
5         C      EACH INPUT REQUEST MUST BE FOLLOWED BY A CALL TO ERCK            G078G      6
          C      REQUIRES ALSO SUBROUTINES ERCK,ERROR78 AND ANNOTA                G078G      7
                 DIMENSION LIST(6)                                               G078G      8
                 LOGICAL ERROR                                                   G078G      9
                 COMMON/ERROR/ERROR                                              G078G     10
10               EXTERNAL ERROR78                                                G078G     11
                 DATA LIST/3*0,3*-1/                                             G078G     12
                 LIST(5)=LOCF(ERROR78)                                           G078G     13
                 CALL SYSTEMC(78,LIST)                                           G078G     14
                 ERROR=.FALSE.                                                   G078G     15
15               END                                                             G078G     16
```

```
                      SUBROUTINE ERCK,RETURNS(A)                                ERCK      2
                C     RETURN(A) IS EXECUTED WHEN A 78 ERROR HAS OCCURRED.       ERCK      3
                      DIMENSION ERRMSG(5)                                       ERCK      4
                      LOGICAL ERROR                                             ERCK      5
     5                COMMON/ERROR/ERROR                                        ERCK      6
                      COMMON /GE/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,          CYBER77   1
                     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO         6JULY77   3
                      DATA ERRMSG/5HINPUT FORMAT ERROR.PICK THIS MSG.THEN REENTER DATA/ ERCK  8
                      IF(.NOT.ERROR)RETURN                                      ERCK      9
    10                CALL ANNOTA(-3.,9.,ERRMSG,50,IDDAD,IBUT,99,-0,-0)         3NOV      4
                 10   CALL GIBUT(0,NCON,IDDT,IDC)                               ERCK      11
                      IF(IDC.NE.99 )GO TO 10                                    ERCK      12
                      CALL GIERAS(IDDAD)                                        ERCK      13
                      ERROR=.FALSE.                                             ERCK      14
    15                RETURN A                                                  ERCK      15
                      END                                                       ERCK      16
```

```
      SUBROUTINE ERROR78                                        ERROR78    2
      ERROR RECOVERY SUBROUTINE                                 ERROR78    3
      LOGICAL ERROR                                             ERROR78    4
      COMMON/ERROR/ERROR                                        ERROR78    5
      ERROR=.TRUE.                                              ERROR78    6
      END                                                       ERROR78    7
```

```
      SUBROUTINE ORIGIN(XORIGIN,YORIGIN)                                ORIGIN     2
C                                                                       ORIGIN     3
C     ORIGIN PROVIDES A REFERENCE POINT ON THE SCREEN OTHER THAN THE    ORIGIN     4
C     DEFAULT REFERENCE POINT. THE DEFAULT REFERENCE POINT IS THE       ORIGIN     5
C     CENTER OF THE SCREEN. THE REFERENCE POINT IS CALLED THE           ORIGIN     6
C     ORIGIN.                                                           ORIGIN     7
C          XORIGIN - I - THE X-COORDINATE IN INCHES OF THE NEW          ORIGIN     8
C                    REFERENCE POINT WITH RESPECT TO THE DEFAULT        ORIGIN     9
C                    REFERENCE POINT.                                   ORIGIN    10
C          YORIGIN - I - THE Y-COORDINATE IN INCHES OF THE NEW          ORIGIN    11
C                    REFERENCE POINT WITH RESPECT TO THE DEFAULT        ORIGIN    12
C                    REFERENCE POINT.                                   ORIGIN    13
C           DGUPIN - O - THE NUMBER OF DIGIGRAPHIC UNITS PER INCH.      ORIGIN    14
C           XORDGU - O - THE X-COORDINATE IN DIGIGRAPHIC UNITS OF       ORIGIN    15
C                    THE NEW REFERENCE POINT WITH RESPECT TO THE        ORIGIN    16
C                    DEFAULT REFERENCE POINT.                           ORIGIN    17
C           YORDGU - O - THE Y-COORDINATE IN DIGIGRAPHIC UNITS OF       ORIGIN    18
C                    THE NEW REFERENCE POINT WITH RESPECT TO THE        ORIGIN    19
C                    DEFAULT REFERENCE POINT.                           ORIGIN    20
C     DGUPIN IS USED IN OTHER ROUTINES TO CONVERT THE COORDINATES OF    ORIGIN    21
C     POINTS FROM INCHES TO DIGIGRAPHIC UNITS. XORDGU AND YORDGU ARE    ORIGIN    22
C     USED TO TRANSLATE THE ORIGIN.                                     ORIGIN    23
C                                                                       ORIGIN    24
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                /PARMSV/   2
      DGUPIN = 200.0                                                    ORIGIN    26
      XORDGU = XORIGIN * DGUPIN                                         ORIGIN    27
      YORDGU = YORIGIN * DGUPIN                                         ORIGIN    28
      RETURN                                                            ORIGIN    29
      END                                                               ORIGIN    30
```

```
      SUBROUTINE PRELIMS                                                PRELIMS    2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/    2
      COMMON/DISKT/I801,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  OMSET(ICOM7+J,I80A,DATA)  STORES DATA(N) IN I80A(1)    /DISKT/     3
      COMMON/TDIS/ITRIM1,PRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,      T30AUG      2
     1XDATMAX,YDATMIN,YDATMAX                                           T30AUG      3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1    10MAY77    61
     1,IDELTF,IDELBC(2)                                                 FLCA02      5
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10) /CROSTB/     2
     1,FMACH,PVALU(10),PNAME(10),IRP(10)                                /CROSTB/    3
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1, /AXBEAD/    2
     X              IAXY2,IAXY3                                         /AXBEAD/    3
      COMMON/AXSBEAD/IAXSXM,IAXSXI,IAXSY1M,IAXSY1I,IAXSY2M,IAXSY2I,     /AXSBEAD/    2
     X               IAXSY3M,IAXSY3I                                    /AXSBEAD/    3
      COMMON/CCMPCOD/IWDZERO,IWDONE,IWDTWO,IWDTHRE,IWDFOUR,IWDFIVE,     /CCMPCOD/    2
     X              IWDSIX,IWDSEVE,IWDEIGH,IWDNINE,IWDTEN               /CCMPCOD/    3
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/    2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/    3
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/    2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77     2
     X               IDINSTR(19),IDDATA8(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT       1
      COMMON/FITBEAD/IFTYPE,IFIWEP,IFIWLP,IFIWI,IFNPARA                 14APR76     1
      COMMON /GP/ IBUF(128),NCON,N8YTE,M8YTE,IGN,ISIN,                  CYB777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77     3
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,    /PGBEAD/    2
     X              IPGLOA                                              /PGBEAD/    3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,      /PLTBEAD/    2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/    3
     X               IPFMN,IPFMV                                        /PLTBEAD/    4
      COMMON/PMBEAD/IPMNEXT,IPMPREV,IPMPLST,IPMMLST,IPMPLOT,IPMPTXT,    /PMBEAD/    2
     X              IPMMTXT                                             /PMBEAD/    3
      COMMON /PLOTCOM/ IFIRST,IPLTFL1,NCPC,ICAL8D,DPLOT(128)            /PLOTCOM/   2
C                                                                       PRELIMS    13
C         ATTACH THE GRAPHICS CONSOLE TO THE JOB                        PRELIMS    14
C                                                                       PRELIMS    15
      NCON=1                                                            PRELIMS    17
      CALL GICNJB(NCON)                                                 PRELIMS    18
C                                                                       6JULY77     7
C        SAVE CURRENT CP TIME                                           6JULY77     8
C                                                                       6JULY77     9
      CPTO=SECOND(CP)                                                   6JULY77    10
C                                                                       PRELIMS    19
C         INITIALIZE THE GRAPHICS PARAMETERS---COMMON/GP/               PRELIMS    20
C                                                                       PRELIMS    21
      N8YTE=0                                                           PRELIMS    22
      M8YTE=380                                                         CYB777      5
      ICODE=1028                                                        PRELIMS    24
      ISTYLE=0                                                          PRELIMS    25
      IBEAM=1                                                           PRELIMS    26
      IGN=1                                                             PRELIMS    27
      ISIN=2                                                            PRELIMS    28
      ISTR=4                                                            PRELIMS    29
      IBUT=8                                                            PRELIMS    30
      IMARK=16                                                          PRELIMS    31
      ICOM=1000                                                         CYB777      6
C                                                                       PRELIMS    32
C         INITIALIZE THE SYSTEM MASK VALUES                             PRELIMS    33
```

```
C                1 - IGNORE MASK                                          PRELIMS   34
C                2 - SINGLE PICK MASK                                     PRELIMS   35
C                4 - STRING PICK MASK                                     PRELIMS   36
C                8 - BUTTON PICK MASK                                     PRELIMS   37
C               16 - MARKER MASK                                          PRELIMS   38
C                                                                         PRELIMS   39
      CALL GIMASK(NCON,1,IGN,IGN)                                         PRELIMS   40
      CALL GIMASK(NCON,2,ISIN,ISIN)                                       14APR76   13
      CALL GIMASK(NCON,4,ISTR,ISTR)                                       14APR76   14
      CALL GIMASK(NCON,8,IBUT,IBUT)                                       14APR76   15
      CALL GIMASK(NCON,C,ISIN + ISTR + IBUT,IMARK)                        14APR76   16
C                                                                         PRELIMS   50
C        SET THE GENERALIZED COMPONENT CODES---COMMON/COMPCOD/            PRELIMS   51
C                                                                         PRELIMS   52
      IWDZERO=100000000006                                                PRELIMS   53
      IWDONE =100000000019                                                PRELIMS   54
      IWDTWO =100000000024                                                PRELIMS   55
      IWDTHRE=100000000039                                                PRELIMS   56
      IWDFOUR=100000000049                                                PRELIMS   57
      IWDFIVE=100000000053                                                PRELIMS   58
      IWDSIX =100000000068                                                PRELIMS   59
      IWDSEVE=100000000078                                                PRELIMS   60
      IWDEIGH=100000000108                                                PRELIMS   61
      IWDNINE=100000000118                                                PRELIMS   62
      IWDTEN =100000000129                                                PRELIMS   63
C                                                                         PRELIMS   64
C        SET THE COMPONENT CODES FOR THE PLOT GROUP BEAD---COMM/PGBEAD/   PRELIMS   65
C                                                                         PRELIMS
      IPGNEXT=IWDZERO+1                                                   PRELIMS   67
      IPGPREV=IWDZERO+2                                                   PRELIMS   68
      IPGTITL=IWDZERO+3                                                   PRELIMS   69
      IPGAXIS=IWDZERO+4                                                   PRELIMS   70
      IPGCTXT=IWDZERO+5                                                   PRELIMS   71
      IPGNCNS=IWDZERO+6                                                   PRELIMS   72
      IPGLCA =IWDZERO+7                                                   PRELIMS   73
C                                                                         PRELIMS   74
C        SET THE COMPONENT CODES FOR THE AXIS BEAD---COMMON/AXBEAD/       PRELIMS   75
C                                                                         PRELIMS   76
      IAXNEXT=IWDZERO+1                                                   PRELIMS   77
      IAXPREV=IWDZERO+2                                                   PRELIMS   78
      IAXATXT=IWDZERO+3                                                   PRELIMS   79
      IAXPP  =IWDZERO+4                                                   PRELIMS   80
      IAXXNME=IWDZERO+5                                                   PRELIMS   81
      IAYNYAX=IWDZERO+6                                                   PRELIMS   82
      IAXY1  =IWDZERO+7                                                   PRELIMS   83
      IAXY2  =IWDZERO+8                                                   PRELIMS   84
      IAXY3  =IWDZERO+9                                                   PRELIMS   85
C                                                                         PRELIMS   86
C        SET THE COMPONENT CODES FOR THE P-M BEAD---COMMON/PMBEAD/        PRELIMS   87
C                                                                         PRELIMS   88
      IPMNEXT=IWDZERO+1                                                   PRELIMS   89
      IPMPREV=IWDZERO+2                                                   PRELIMS   90
      IPMPLST=IWDZERO+3                                                   PRELIMS   91
      IPMMLST=IWDZERO+4                                                   PRELIMS   92
      IPMPLOT=IWDZERO+5                                                   PRELIMS   93
      IPMATXT=IWDZERO+6                                                   PRELIMS   94
      IPMMTXT=IWDZERO+7                                                   PRELIMS   95
```

```
      C                                                                 PRELIMS   96
      C     SET THE COMPONENT CODES FOR THE PLOT BEAD---COMMON/PLTBEAD/ PRELIMS   97
      C                                                                 PRELIMS   98
            IPNEXTC=IWDZERO+1                                           PRELIMS   99
            IPNEXTT=IWDZERO+2                                           PRELIMS  100
            IPPREVC=IWDZERO+3                                           PRELIMS  101
            IPPREVT=IWDZERO+4                                           PRELIMS  102
            IPPG   =IWDZERO+5                                           PRELIMS  103
            IPTITLE=IWDZERO+6                                           PRELIMS  104
            IPAXIS =IWDZERO+7                                           PRELIMS  105
            IPAXISS=IWDZERO+8                                           PRELIMS  106
            IPFM   =IWDZERO+9                                           PRELIMS  107
            IPLABEL=IWDZERO+10                                          PRELIMS  108
            IPCURVE=IWDZERO+11                                          PRELIMS  109
            IPNMULT=IWDZERO+12                                          PRELIMS  110
            IPFMX  =IWDZERO+13                                          PRELIMS  111
            IPFMY  =IWDZERO+14                                          PRELIMS  112
      C                                                                 PRELIMS  113
      C     SET THE COMPONENT CODES FOR THE CURVE BEAD---COMMON/CRVBEAD/ PRELIMS  114
      C                                                                 PRELIMS  115
            ICNEXT =IWDZERO+1                                           PRELIMS  116
            ICPREV =IWDZERO+2                                           PRELIMS  117
            ICPARAM=IWDZERO+3                                           PRELIMS  118
            ICPARAV=IWDZERO+4                                           PRELIMS  119
            ICNPTS =IWDZERO+5                                           PRELIMS  120
            ICSYM  =IWDZERO+6                                           PRELIMS  121
            ICPEN  =IWDZERO+7                                           PRELIMS  122
            ICXPTS =IWDZERO+8                                           PRELIMS  123
            ICYPTS =IWDZERO+9                                           PRELIMS  124
            ICINCFP=IWDZERO+10                                          PRELIMS  125
            ICFIT  =IWDZERO+11                                          PRELIMS  126
            ICSTAT =IWDZERO+12                                          PRELIMS  127
      C                                                                 PRELIMS  128
      C     SET THE COMPONENT CODES FOR THE FIT BEAD---COMMON/FITBEAD/  PRELIMS  129
      C                                                                 PRELIMS  130
            IFTYPE =IWDZERO+1                                           PRELIMS  131
            IFINVFP=IWDZERO+2                                           PRELIMS  132
            IFIVVLP=IWDZERO+3                                           PRELIMS  133
            IFIVI  =IWDZERO+4                                           PRELIMS  134
            IFNPARA=IWDZERO+5                                           PRELIMS  135
      C                                                                 PRELIMS  136
      C     SET THE COMP CODES FOR THE AXIS SCALE BEAD---COMMON/AXSBEAD/ PRELIMS  137
      C                                                                 PRELIMS  138
            IAXSXM =IWDZERO+1                                           PRELIMS  139
            IAXSXI =IWDZERO+2                                           PRELIMS  140
            IAXSY1M=IWDZERO+3                                           PRELIMS  141
            IAXSY1I=IWDZERO+4                                           PRELIMS  142
            IAXSY2M=IWDZERO+5                                           PRELIMS  143
            IAXSY2I=IWDZERO+6                                           PRELIMS  144
            IAXSY3M=IWDZERO+7                                           PRELIMS  145
            IAXSY3I=IWDZERO+8                                           PRELIMS  146
      C                                                                 14APR76   17
      C     INITIALIZE THE STATUS WORDS (FOR THE CURVE FITS) IN THE CURVE 14APR76   18
      C     DISPLAY ACCESS TABLE TO INACTIVE.                           14APR76   19
      C                                                                 14APR76   20
            DO 10 I=1,20                                                14APR76   21
            ICDATAB(I,1,2) = 8HINACTIVE                                 14APR76   22
```

```
                10 CONTINUE                                                      14APR76    23
                C                                                                ADPTPRE     8
                C       SET ALL OF IODAD TO ZERO                                 ADPTPRE     9
      175       C                                                                ADPTPRE    10
                   DO 20 I=1,80                                                  ADPTPRE    11
                20 IODAC(I)=0                                                    ADPTPRE    12
                C                                                                INITR       5
                C       INITIALIZE VALUES IN ITRINT AND DISKT COMMON AREAS       INITR       6
      180       C                                                                INITR       7
                   ITFIM=0                                                       INITR       8
                   N6=6                                                          INITR       9
                   ICOM1=100000000009                                            INITR      11
                   ICOM2=700000000018                                            INITR      12
      185          NX=1                                                         FIXDUM      9
                   NY=2                                                         FIXDUM     10
                   XMN=YMN=0.                                                   FIXDUM     11
                   DELX=.5                                                      FIXDUM     12
                   DELY=.2                                                      FIXDUM     13
      190          ITFIM1=0                                                     FIXDUM     14
                   IVC=1HX                                                      FIXDUM     15
                   XM=-.03                                                      FIXDUM     16
                   JT=1                                                         FIXDUM     17
                   CALL GOTAG                                                   50CT       6
      195          CTYPE(1)=10HCOORDINATE                                       FIXDUM     18
                   CTYPE(2)=6HSLOPES                                            FIXDUM     19
                   CTYPE(3)=7HMINIMUM                                           FIXDUM     20
                   CTYPE(4)=5HDELTA                                             FIXDUM     21
                   CTYPE(5)=7HMAXIMUM                                           10MAY77    1
      200          TYPE=10HCOORDINATE                                           FIXDUM     22
                   ICOSAVE=0                                                    FIXDUM     23
                   IFIRST=0                                                     25AUG      3
                   IPLTFL1=0                                                    25AUG      4
                   RETURN                                                       PRELIMS  147
      205          END                                                          PRELIMS  148
```

191

```
      SUBROUTINE SQUARE(XBEG,YBEG,XLEN,YLEN,IODAD,IODT,IODC)                SQUARE      2
C                                                                           SQUARE      3
C         THIS ROUTINE DISPLAYS THE RECTANGLE DEFINED BY THE                SQUARE      4
C         FIRST FOUR PARAMETERS IN THE CALLING SEQUENCE.                    SQUARE      5
C              XBEG - I - THE X-COORDINATE IN INCHES OF THE                 SQUARE      6
C                         LOWER LEFT CORNER OF THE RECTANGLE.               SQUARE      7
C              YBEG - I - THE Y-COORDINATE IN INCHES OF THE                 SQUARE      8
C                         LOWER LEFT CORNER OF THE RECTANGLE.               SQUARE      9
C              XLEN - I - THE HORIZONTAL LENGTH OF THE RECTANGLE.           SQUARE     10
C              YLEN - I - THE VERTICAL HEIGHT OF THE RECTANGLE.             SQUARE     11
C             IODAD - O - THE SYSTEM DEFINED ASSOCIATIVE ADDRESS            SQUARE     12
C                         OF THE DISPLAY ITEM.                              SQUARE     13
C              IODT - I - THE DISPLAY TYPE.                                 SQUARE     14
C              IODC - I - THE DISPLAY CODE.                                 SQUARE     15
C                                                                           SQUARE     16
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                      CY8777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                     6JULY77     3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                    /PARMSV/    2
      IXBEG = IFIX(XBEG * DGUPIN + XORDGU + .5)                             SQUARE     19
      IYBEG = IFIX(YBEG * DGUPIN + YORDGU + .5)                             SQUARE     20
      IXEND = IFIX((XBEG + XLEN) * DGUPIN + XORDGU + .5)                    SQUARE     21
      IYEND = IFIX((YBEG + YLEN) * DGUPIN + YORDGU + .5)                    SQUARE     22
      NBYTE = 0                                                             SQUARE     23
      CALL GURSET(IXBEG,IYBEG,ICODE,IBUF,NBYTE,MBYTE)                       SQUARE     24
      CALL GUSEGT(IXBEG,IYBEG,ISTYLE,IBUF,NBYTE,MBYTE)                      SQUARE     25
      CALL GUSEG(IXEND,IYBEG,1)                                             SQUARE     26
      CALL GUSEG(IXEND,IYEND,1)                                             SQUARE
      CALL GUSEG(IXBEG,IYEND,1)                                             SQUARE
      CALL GUSEG(IXBEG,IYBEG,1)                                             SQUARE     29
      CALL GIDISP(NCON,IBUF,NBYTE,IODAD,IODT,IODC)                          SQUARE     30
      RETURN                                                                SQUARE     31
      END                                                                   SQUARE     32
```

```
1                SUBROUTINE STOP                                    STOP      2
                 CALL AETSKC (6HSTOPIT)                             25AUG     5
                 END                                                STOP      9
```

```
      SUBROUTINE TERASE(I,K)                                                TERASE     2
C                                                                           TERASE     3
C         TERASE ERASES THE FIRST K DISPLAYS WHOSE ASSOCIATIVE ADDRESSES    TERASE     4
C         ARE CONTAINED IN ARRAY I. THE MAXIMUM VALUE OF K IS 100.          TERASE     5
C                                                                           TERASE     6
      DIMENSION I(K)                                                        TERASE     7
      INTEGER TEMP                                                          TERASE     8
      IF(K .GE. 100) GO TO 10                                               TERASE     9
      TEMP = I(K+1)                                                         TERASE    10
      I(K+1) = -9                                                           TERASE    11
   10 CALL GIERAS(I(1),I(2),I(3),I(4),I(5),I(6),I(7),I(8),I(9),I(10),       TERASE    12
     X            I(11),I(12),I(13),I(14),I(15),I(16),I(17),I(18),I(19),   TERASE    13
     X            I(20),I(21),I(22),I(23),I(24),I(25),I(26),I(27),I(28),   TERASE    14
     X            I(29),I(30),I(31),I(32),I(33),I(34),I(35),I(36))         TERASE    15
      IF(K .LE. 36) GO TO 99                                                TERASE    16
      CALL GIERAS(I(37),I(38),I(39),I(40),I(41),I(42),I(43),I(44),I(45),    TERASE    17
     X            I(46),I(47),I(48),I(49),I(50),I(51),I(52),I(53),I(54),   TERASE    18
     X            I(55),I(56),I(57),I(58),I(59),I(60),I(61),I(62),I(63),   TERASE    19
     X            I(64),I(65),I(66),I(67),I(68),I(69),I(70),I(71),I(72))   TERASE    20
      IF(K .LE. 72) GO TO 99                                                TERASE    21
      CALL GIERAS(I(73),I(74),I(75),I(76),I(77),I(78),I(79),I(80),I(81),    TERASE    22
     X            I(82),I(83),I(84),I(85),I(86),I(87),I(88),I(89),I(90),   TERASE    23
     X            I(91),I(92),I(93),I(94),I(95),I(96),I(97),I(98),I(99),   TERASE    24
     X            I(100))                                                  TERASE    25
      IF(K .GE. 100) RETURN                                                 TERASE    26
   99 I(K+1) = TEMP                                                         TERASE    27
      RETURN                                                                TERASE    28
      END                                                                   TERASE    29
```

```
      OVERLAY(1,0)                                                          INTRO    2
      PROGRAM INTRO                                                         INTRO    3
      COMMON/DISPLAY/IOSTOP,IODAD(50),NN,NNEND(10),IOSPEC(19),IOCF(41),     /DISPLAY/ 2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),            6JULY77  2
     Y               IOINSTR(19),ICOATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM   1SEPT    1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                      CYBER77  1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                     6JULY77  3
      CALL ORIGIN(-2.64,3.6)                                                INTRO    6
      CALL ANNOTA (.36,-.2,44HPLOTTING LANGUAGE UTILIZING GRAPHICAL OPTI    NEWFACE  1
     1CS,44,IODAD(1),IGN,-0,-0,-0)                                          NEWFACE  2
      CALL ANNOTA (2.86,-.4,7H(PLUGO),7,IODAD(2),IGN,-0,-0,-0)              NEWFACE  3
C                                                                           NEWFACE  4
      CALL ANNOTA (0.,-1.2,12HPREPARED BY:,12,IODAD(3),IGN,-0,-0,-0)        NEWFACE  5
      CALL ANNOTA (.72,-1.8,21HFLIGHT VEHICLE BRANCH,21,IODAD(4),IGN,       NEWFACE  6
     1    -0,-0,-0)                                                        NEWFACE  7
      CALL ANNOTA (.72,-2.0,22HAEROMECHANICS DIVISION,22,IODAD(5),IGN,      NEWFACE  8
     1    -0,-0,-0)                                                        NEWFACE  9
      CALL ANNOTA (.72,-2.2,36HAIR FORCE FLIGHT DYNAMICS LABORATORY,36,     NEWFACE 10
     1    IODAD(6),IGN,-0,-0,-0)                                           NEWFACE 11
C                                                                           INTRO   19
      CALL ANNOTA(0.,-2.8,20HIN COOPERATION WITH:,20,IODAD(7),IGN,-0,-0,    INTRO   20
     X            -0)                                                      INTRO   21
      CALL ANNOTA(.72,-3.4,26HDIGITAL PROGRAMMING BRANCH,26,IODAD(8),       INTRO   22
     X            IGN,-0,-0,-0)                                            INTRO   23
      CALL ANNOTA(.72,-3.6,27HDIGITAL PROCESSING DIVISION,27,IODAD(9),      INTRO   24
     X            IGN,-0,-0,-0)                                            INTRO   25
      CALL ANNOTA (.72,-3.8,30HAIR FORCE AERONAUTICAL SYSTEMS,30,           NEWFACE 13
     1    IODAD(10),IGN,-0,-0,-0)                                          NEWFACE 14
      CALL ANNOTA (.72,-4.0,24HDIVISION COMPUTER CENTER,24,                 NEWFACE 15
     1    IODAD(11),IGN,-0,-0,-0)                                          NEWFACE 16
C                                                                           INTRO   30
      CALL ANNOTA(0.,-4.6,33HPROJECT ENGINEER---JAMES L. TERRY,33,          INTRO   31
     X            IODAD(12),IGN,-0,-0,-0)                                  INTRO   32
      CALL ANNOTA(0.,-5.2,29HPROGRAMMERS---WAYNE A. JANSEN,29,IODAD(13),    INTRO   33
     X            IGN,-0,-0,-0)                                            INTRO   34
      CALL ANNOTA(1.68,-5.4,21HCAPT. STEPHEN E. HAMM,21,IODAD(14),IGN,      INTRO   35
     X            -0,-0,-0)                                                INTRO   36
      CALL ANNOTA (1.68,-5.6,15HSHARON J. FOLEY,15,IODAD(15),               NEWFACE 17
     1    IGN,-0,-0,-0)                                                    NEWFACE 18
      CALL ANNOTA (1.68,-5.8,17HTHOMAS S. ROWLAND,17,IODAD(16),             NEWFACE 19
     1    IGN,-0,-0,-0)                                                    NEWFACE 20
C                                                                           NEWFACE 21
      CALL ANNOTA (0.,-6.4,26HDOCUMENTED IN AFFDL TR-77-,26,                NEWFACE 22
     1    IODAD(17),IGN,-0,-0,-0)                                          NEWFACE 23
      CALL ANNOTA (2.52,-7.6,7HEXECUTE,7,IODAD(18),IBUT,-0,-0,-0)           NEWFACE 24
C                                                                           INTRO   41
      CALL GIBUT(0,NCON,IOOT)                                               INTRO   42
      NN=18                                                                 NEWFACE 25
      CALL TERASE(IODAD(1),NN)                                              INTRO   44
      NN=0                                                                  INTRO   45
      CALL RETRN                                                            INTRO   46
      END                                                                   INTRO   47
```

```
      OVERLAY(2,0)                                                      PGSPECS    2
      PROGRAM PGSPECS                                                   PGSPECS    3
C                                                                       PGSPECS    4
C         THIS PROGRAM MANAGES THE DISPLAY OF THE INSTRUCTIONS, OPTIONS, PGSPECS    5
C         AND PLOT GROUP SPECIFICATIONS. ONCE THE SPECIFICATIONS ARE    PGSPECS    6
C         DISPLAYED, THE ROUTINE PROCESSES THE CHOICE OF SPECIFICATIONS PGSPECS    7
C         AND OPTION.                                                   PGSPECS    8
C                                                                       PGSPECS    9
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),IDDATA8(20,4,2),NDIP,IDBLOCK(20),NDIPM 1SEPT     1
      COMMON /GP/ IBUF(126),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBER77    1
     X     IDTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/OPTION/IOPTION,ISHINE                                      LASTBUX    1
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,    /PGBEAD/    2
     X              IPGLOA                                             /PGBEAD/    3
    1 CALL INSTRUC                                                      PGSPECS   15
      CALL OPTIONS                                                      PGSPECS   16
   10 CALL SPECS                                                        5OCT       7
C                                                                       PGSPECS   18
C         WAIT FOR A BUTTON PICK FROM THE FOLLOWING MENU.               PGSPECS   19
C                    1 - CROSS PLOT                                     PGSPECS   20
C                    2 - TRIM                                           PGSPECS   21
C                    3 - NEXT PLOT GROUP                                PGSPECS   22
C               4 - PREVIOUS PLOT GROUP                                 3MARCH     1
C                  999 - STOP                                           PGSPECS   23
C                                                                       PGSPECS   24
   20 CALL GIBUT(0,NCON,IODT,IODC,IPLTPTR)                              PGSPECS   25
      IF(IODC .EQ. 1) GO TO 30                                          PGSPECS   26
      IF(IODC .EQ. 2) GO TO 40                                          PGSPECS   27
      IF(IODC .EQ. 3) GO TO 50                                          PGSPECS   28
      IF(IODC.EQ.4)GO TO 55                                             3MARCH     2
      IF(IODC .EQ. 254) CALL STOP                                       14APR76   25
      GO TO 20                                                          PGSPECS   30
C                                                                       PGSPECS   31
C         CROSS PLOT WAS SELECTED.                                      PGSPECS   32
C             1 - ERASE THE MENU, INSTRUCTIONS, AND PLOT SPECIFICATIONS. PGSPECS   33
C             2 - RETRIEVE THE POINTER TO THE FIRST PLOT IN THE CROSS   PGSPECS   34
C                 PLOT SEQUENCE FROM THE IDWA OF THE PICKED CARD IMAGE. PGSPECS   35
C             3 - CALL THE ROUTINE TO DISPLAY THE FIRST PLOT. PASS THE  PGSPECS   36
C                 POINTER TO THE FIRST PLOT THROUGH COMMON/BEADS/.      PGSPECS   37
C                                                                       PGSPECS   38
   30 IOPTION=1HCROSS PLOT                                              PGSPECS   39
      CALL TERASE(IDDAD(1),NN)                                          PGSPECS   40
      NN=0                                                              PGSPECS   41
      CALL GIFIB(NCON,IODT,IODC,IPLTPTR)                                PGSPECS   42
      IF(IODT.NE.0)GO TO 41                                             5OCT       8
C                                                                       PGSPECS   44
C         ERROR---NO P-M GROUP HAS BEEN SELECTED.                       PGSPECS   45
C                                                                       PGSPECS   46
      CALL ERRORM( 4.0,2.5,54HYOU HAVE NOT SELECTED A P-M GROUP TO BE CR 14MARCH   13
     XOSS PLOTTED.,54,7HPGSPECS,6HINFORM)                               PGSPECS   48
      GO TO 1                                                           PGSPECS   49
C                                                                       PGSPECS   50
C         TRIM WAS SELECTED.                                            PGSPECS   51
C             1 - ERASE THE MENU, INSTRUCTIONS, AND PLOT SPECIFICATIONS. PGSPECS   52
```

```
      C              2 - CALL THE ROUTINE TO DISPLAY THE FIRST PLOT IN THE TRIM PGSPECS    53
      C                  SEQUENCE. THE POINTER TO THE FIRST PLOT WAS RETRIEVED  PGSPECS    54
   60 C                  IN THE CALL TO GIBUT. PASS IT THROUGH COMMON/BEADS/ TO PGSPECS    55
      C                  THE PLOTTING ROUTINE.                                  PGSPECS    56
      C                                                                         PGSPECS    57
         40 IOPTION=4HTRI*                                                      PGSPECS    58
            CALL TERASE(IOOAD(1),NN)                                            PGSPECS    59
   65       NN=0                                                                PGSPECS    60
         41 CONTINUE                                                            5OCT       9
            CALL GETSKC(7HFRSTPL2)                                              10MAY77    2
      C                                                                         PGSPECS    62
      C         NEXT PLOT GROUP WAS SELECTED                                    PGSPECS    63
   70 C                                                                         PGSPECS    64
         50 CALL DMGET(IPGNEXT,IPGPTR,IBEAD)                                    PGSPECS    65
            GO TO 56                                                            3MARCH     3
      C                                                                         3MARCH     4
      C              PREVIOUS PLOT GROUP WAS SELECTED                           3MARCH     5
   75 C                                                                         3MARCH     6
         55 CALL DMGET(IPGPREV,IPGPTR,IBEAD)                                    3MARCH     7
         56 CONTINUE                                                            3MARCH     8
            IF(IBEAD .EQ. 0) GO TO 60                                           PGSPECS    66
            IPGPTR=IBEAD                                                        PGSPECS    67
   80       N=NNEND(2)-NNEND(1)                                                 PGSPECS    68
            CALL TERASE(IOOAD(NNEND(1)+1),N)                                    PGSPECS    69
            NN=NNEND(1)                                                         PGSPECS    70
            GO TO 10                                                            PGSPECS    71
      C                                                                         PGSPECS    72
   85 C         ERROR---THERE ARE NO MORE PLOT GROUPS.                          PGSPECS    73
      C                                                                         PGSPECS    74
         60 CALL ERRORM(8.,2.5,30HTHERE ARE NO MORE PLOT GROUPS.,30,7HPGSPECS, PGSPECS    75
           *              6HINFORM)                                             PGSPECS    76
            GO TO 20                                                            PGSPECS    77
   90 C                                                                         PGSPECS    78
            END                                                                 PGSPECS    79
```

```
      SUBROUTINE CARD(IBEAD,ITYPE,X,Y,IODT,IOWA)                                CARD      2
C                                                                               CARD      3
C     *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *         CARD      4
C         THIS SUBROUTINE:                                                      CARD      5
C           1. BUILDS A RECORD TO BE DISPLAYED BY MERGING THE CARD TYPE         CARD      6
C              (AXIS, TITLE, CONDITION, PARAMETER, OR MULTIPLIER) AND           CARD      7
C              THE TEXT INFORMATION(FUNCTION AND ARGUMENTS); AND                CARD      8
C                                                                               CARD      9
C           2. DISPLAYS THE RECORD.                                             CARD     10
C                                                                               CARD     11
C         THE SUBROUTINE PARAMETERS ARE:                                        CARD     12
C           1. IBEAD   - INPUT - ADDRESS OF THE BEAD WHICH CONTAINS THE         CARD     13
C                        TEXT INFORMATION.                                      CARD     14
C           2. ITYPE   - INPUT - A HOLLERITH WORD CONTAINING "AXIS",            CARD     15
C                        "TITLE", "CONDITION", "PARAMETER", OR                  CARD     16
C                        "MULTIPLIER".                                          CARD     17
C           3. X,Y     - INPUT AND OUTPUT - COORDINATES OF THE POINT ON         CARD     18
C                        THE SCREEN AT WHICH THE CARD IS TO BE                  CARD     19
C                        DISPLAYED.                                             CARD     20
C           4. IODT    - INPUT - DISPLAY TYPE OF THE CARD.                      CARD     21
C           5. IOWA    - INPUT - IF THE CARD IS TO BE NONPICKABLE, THEN         CARD     22
C                        IOWA=-0. IF THE CARD IS PICKABLE, THEN IOWA IS         CARD     23
C                        THE ADDRESS OF THE FIRST PLOT BEAD IN THE PM           CARD     24
C                        GROUP WHICH THE CARD PARTIALLY DESCRIBES.              CARD     25
C     *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *         CARD     26
C                                                                               CARD     27
      COMMON/BLOCK/IBLOCK(50)                                                   /BLOCK/
      COMMON/COMPCOD/IWOZERO,IWOONE,IWOTWO,IWOTHRE,IWOFOUR,IWOFIVE,             /COMPCOD/
     X               IWOSIX,IWOSEVE,IWOEIGH,IWONINE,IWOTEN                      /COMPCOD/
      COMMON/DISPLAY/IOSTOP,IOOAO(80),NN,NNEND(10),IOSPEC(19),IOCF(41),         /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),                  6JULY77    2
     X               IOINSTR(19),IOOATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM         1SEPT      1
      DIMENSION IFANOA(3), ICARD(9)                                             CARD      31
      EQUIVALENCE (IFANOA,IBLOCK), (ICARD,IBLOCK(9))                            CARD      32
C                                                                               CARD      33
C         MOVE THE CONTENTS OF THE TEXT BEAD INTO THE IFANDA ARRAY.             CARD      34
C                                                                               CARD      35
      CALL OMGET(IWOTWO,IBEAO,NCHAR)                                            CARD      36
      CALL OMGET(7C01000000038,IBEAD,IFANOA(1))                                 CARD      37
C                                                                               CARD      38
C         MERGE THE CARD TYPE (A, T, C, P, OR M) WITH THE                       CARD      39
C         FUNCTION AND ARGUMENTS.                                               CARD      40
C                                                                               CARD      41
      ENCODE(82,1000,ICARD) ITYPE, IFANDA                                       CARD      42
 1000 FORMAT(A1,1X,8A10)                                                        CARD      43
C                                                                               CARD      44
C         DISPLAY THE CARD IMAGE.                                               CARD      45
C                                                                               CARD      46
      NCHAR = NCHAR + 2                                                         CARD      47
      NN = NN + 1                                                               CARD      48
      CALL ANNOTA(X,Y,ICARD,NCHAR,IOOAO(NN),IOOT,0,IOWA,-0)                     14APR76    26
      RETURN                                                                    CARD      50
      END                                                                       CARD      51
```

```
      SUBROUTINE INSTRUC                                                INSTRUC    2
      COMMON/DISPLAY/IOSTOP,IODAD(81),NN,NNEND(10),IOSPEC(19),IOCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IOTITLE(4),IOCD(23),IOMENU(20),          6JULY77    2
     X               IDINSTR(19),ICOATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYB777     1
     X     JSTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
C                                                                       INSTRUC    5
C     PURPOSE---TO DISPLAY THE INSTRUCTIONS WHICH EXPLAIN HOW TO        INSTRUC    6
C               CHOOSE THE DATA WHICH IS TO BE MANIPULATED.             INSTRUC    7
C                                                                       INSTRUC    8
      CALL ORIGIN(-6.,7.)                                               INSTRUC    9
      CALL SQUARE(0.,-5.5,6.52,5.5,IODAD(1),IGN,-0)                     INSTRUC   10
      CALL ANNOTA(2.4,.25,12HINSTRUCTIONS,12,IODAD(2),IGN,-0,-0,-0)     INSTRUC   11
      CALL ANNOTA(.5,-1.,47H1. IF YOU ARE GOING TO TRIM THE SPECIFIED DA INSTRUC   12
     XTA,,47,IODAD(3),IGN,-0,-0,-0)                                     INSTRUC   13
      CALL ANNOTA(.66,-1.5,13HPICK TRIM.,10,IODAD(4),IGN,-0,-0,-0)      INSTRUC   14
      CALL ANNOTA(.5,-2.5,43H2. IF YOU ARE GOING TO CROSS PLOT THE DATA: INSTRUC   15
     X           ,43,IODAD(5),IGN,-0,-0,-0)                             INSTRUC   16
      CALL ANNOTA(1.22,-3.5,41HA. PICK ANY CARD IMAGE WHICH IS UNIQUE TO INSTRUC   17
     X           ,41,IODAD(6),IGN,-0,-0,-0)                             INSTRUC   18
      CALL ANNOTA(1.56,-4.,38HTHE DESCRIPTION OF THE PERTINENT DATA.,38, INSTRUC   19
     X           IODAD(7),IGN,-0,-0,-0)                                 INSTRUC   20
      CALL ANNOTA(1.22,-5.,19HB. PICK CROSS PLOT.,19,IODAD(8),IGN,-0,-0, INSTRUC   21
     X           -0)                                                    INSTRUC   22
      RETURN                                                            INSTRUC   23
      END                                                               INSTRUC   24
```

```
      SUBROUTINE OPTIONS                                                    OPTIONS     2
C                                                                           OPTIONS     3
C         PURPOSE---TO DISPLAY THE OPTIONS MENU                             OPTIONS     4
C                                                                           OPTIONS     5
      COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNEND(10),IOSPEC(19),IOCF(41),     /DISPLAY/   2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),            6JULY77     2
     X               IOINSTR(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM   1SEPT       1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                    CYBER77     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                    6JULY77     3
      CALL ORIGIN(3.2,7.)                                                   OPTIONS     8
      CALL SQUARE (0.,-4.5,2.3,4.5,IODAD(9),IGN,-0)                         14MARCH    14
      CALL ANNOTA(.74,.85,12HTASK OPTIONS,12,IODAD(10),IGN,-0,-0,-0)        OPTIONS    10
      CALL ANNOTA(.5,-1.,19HCROSS PLOT,10,IODAD(11),IBUT,1,-0,-0)           OPTIONS    11
      CALL ANNOTA (.5,-2.,4HTRIM,4,IODAD(12),IBUT,2,4HTRIM,-0)              3AUG        1
      CALL ANNOTA(.5,-3.,15HNEXT PLOT GROUP,15,IODAD(13),IBUT,3,-0,-0)      OPTIONS    13
      CALL ANNOTA(.5,-4.,19HPREVIOUS PLOT GROUP,19,IODAD(14),IBUT,4,-0,     3MARCH      9
     1-0)                                                                   3MARCH     10
      NN=14                                                                 3MARCH     11
      NNEND(1)=NN                                                           OPTIONS    15
      RETURN                                                                OPTIONS    16
      END                                                                   OPTIONS    17
```

```
      SUBROUTINE SPECS                                                  50CT    20
C                                                                       SPECS    3
C     THIS ROUTINE DISPLAYS THE PLOT SPECIFICATIONS FOR THE CURRENT     SPECS    4
C     PLOT GROUP. (ONE PLOT GROUP AT A TIME.) THE CURRENT PLOT GROUP    SPECS    5
C     IS THE PLOT GROUP WHOSE PLOT GROUP BEAD POINTER IS STORED IN      SPECS    6
C     WORD "IPGPTR".                                                    SPECS    7
C                              *** CROSS PLOT ***                       SPECS    8
C     IF A SPECIFICATION IS UNIQUE TO A GIVEN P-M GROUP, THEN IT IS     SPECS    9
C     DISPLAYED AS A SINGLE PICK ITEM AND IT CAN BE PICKED TO           SPECS   10
C     DETERMINE THE P-M GROUP WHICH IS TO BE CROSS PLOTTED. IF IT IS    SPECS   11
C     NOT UNIQUE, THEN IT IS DISPLAYED AS A NONPICKABLE ITEM.           SPECS   12
C                                  *** TRIM ***                         SPECS   13
C     IF THE TRIM OPTION IS PICKED, THEN THE PICKABILITY OF THE         SPECS   14
C     INDIVIDUAL PLOT SPECIFICATIONS IS IRRELEVANT. THE ONLY            SPECS   15
C     PERTINENT INFORMATION IS THE ADDRESS OF THE FIRST PLOT BEAD IN    SPECS   16
C     THE PLOT GROUP. THIS ADDRESS IS STORED IN THE "IDWA" OF THE       SPECS   17
C     TRIM BUTTON.                                                      SPECS   18
C                                                                       SPECS   19
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1, /AXBEAD/   2
     X              IAXY2,IAXY3                                         /AXBEAD/   3
      COMMON/COMPCOD/IWDZERO,IWDONE,IWDTWO,IWDTHRE,IWDFOUR,IWDFIVE,     /COMPCOD/   2
     X               IWDSIX,IWDSEVE,IWDEIGH,IWDNINE,IWDTEN              /COMPCOD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77     2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPW 1SEPT       1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYB777      1
     X     ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,    /PGBEAD/   2
     X              IPGLCA                                              /PGBEAD/   3
      COMMON/PMBEAD/IPMNEXT,IPMPREV,IPMPLST,IPMMLST,IPMPLOT,IPMPTXT,    /PMBEAD/   2
     X              IPMMTXT                                             /PMBEAD/   3
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,      /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/   3
     X               IPFMN,IPFMV                                        /PLTBEAD/   4
      LOGICAL MULTPM                                                    SPECS      26
      CALL ORIGIN(-6.,0.)                                               SPECS      27
C                                                                       SPECS      28
C     GET POINTERS TO THE "FIRST" AXIS, PM, AND PLOT BEADS.             SPECS      29
C                                                                       SPECS      30
      CALL DMGET(IPGAXIS,IPGPTR,IAXPTR)                                 SPECS      31
      CALL DMGET(IAXPM,IAXPTR,IPMPTR)                                   SPECS      32
      CALL DMGET(IPMPLOT,IPMPTR,IPLTPTR)                                SPECS      33
C                                                                       50CT       23
C     CHASE YOUR TAIL TO GET BACK THE ORIGINAL PLOT SPECS               50CT       24
C                                                                       50CT       25
      CALL DMGET (IPPG,IPLTPTR,IPGPTR)                                  50CT       26
      CALL DMGET (IPGAXIS,IPGPTR,IAXPTR)                                50CT       27
      CALL DMGET (IAXPM,IAXPTR,IPMPTR)                                  50CT       28
C                                                                       SPECS      34
C     STORE THE POINTER TO THE FIRST PLOT BEAD IN THE IDWA OF THE       SPECS      35
C     TRIM BUTTON'S INFORMATION BLOCK.                                  SPECS      36
C                                                                       SPECS      37
      CALL GIMOVE(-0,-0,-0,IDDAD(12),-0,-0,IPLTPTR,-0)                  SPECS      38
C                                                                       SPECS      39
C     BEGIN THE DISPLAY OF THE PLOT SPECIFICATIONS.                     SPECS      40
C                                                                       SPECS      41
```

```
                  Y=-.5                                                    SPECS     42
                  NN=NN+1                                                  SPECS     43
   60             CALL ANNOTA(.5,Y,11H PLOT GROUP,11,IDDAD(NN),IGN,-0,-0,-0)  SPECS  44
                  C                                                        SPECS     45
                  C     IF TITLE CARDS EXIST, DISPLAY THE FIRST ONE.       SPECS     46
                  C                                                        SPECS     47
                  CALL DMGET(IPGTITL,IPGPTR,ITPTR)                         SPECS     48
   65             IF(ITPTR .EQ. 0) GO TO 20                                SPECS     49
                10 Y=Y-.5                                                  SPECS     50
                  CALL CARD(ITPTR,5HTITLE,.5,Y,IGN,-0)                     SPECS     51
                  C                                                        SPECS     52
                  C     CHECK FOR ADDITIONAL TITLE CARDS.                  SPECS     53
   70             C                                                        SPECS     54
                  CALL DMGET(IWCONE,ITPTR,ITPTR)                           SPECS     55
                  IF(ITPTR .EQ. 0) GO TO 20                                SPECS     56
                  GO TO 10                                                 SPECS     57
                  C                                                        SPECS     58
   75             C     IF CONDITION CARDS EXIST, DISPLAY THE FIRST ONE.   SPECS     59
                  C                                                        SPECS     60
                20 CALL DMGET(IPGCTXT,IPGPTR,ICONPTR)                      SPECS     61
                  IF(ICONPTR .EQ. 0) GO TO 40                              SPECS     62
                30 Y=Y-.5                                                  SPECS     63
   80             CALL CARD(ICONPTR,9HCONDITION,.5,Y,IGN,-0)               SPECS     64
                  C                                                        SPECS     65
                  C     CHECK FOR ADDITIONAL CONDITION CARDS.              SPECS     66
                  C                                                        SPECS     67
                  CALL DMGET(IWCONE,ICONPTR,ICONPTR)                       SPECS     68
   85             IF(ICONPTR .EQ. 0) GO TO 40                              SPECS     69
                  GO TO 30                                                 SPECS     70
                  C                                                        SPECS     71
                  C     DISPLAY THE AXIS CARD---A DETERMINATION MUST BE MADE WHETHER  SPECS  72
                  C     THE AXIS CARD WILL BE PICKABLE. THIS IS DONE BY EXAMINING FOR  SPECS  73
   90             C     MORE THAN ONE PM GROUP IN THE AXIS GROUP. IF THERE IS MORE  SPECS  74
                  C     THAN ONE PM GROUP, THEN THE AXIS CARD IS NONPICKABLE.  SPECS  75
                  C     OTHERWISE, IT IS PICKABLE.                         SPECS     76
                  C                                                        SPECS     77
                40 CALL DMGET(IAXATXT,IAXPTR,IAPTR)                        SPECS     78
   95             MULTPM=.FALSE.                                           SPECS     79
                  CALL DMGET(IPMNEXT,IPMPTR,INXTPM)                        SPECS     80
                  Y=Y-.5                                                   SPECS     81
                  IF(INXTPM .NE. 0) MULTPM=.TRUE.                          SPECS     82
                  IF(MULTPM) GO TO 50                                      SPECS     83
  100             CALL CARD(IAPTR,4HAXIS,.5,Y,ISIN,IPLTPTR)                SPECS     84
                  GO TO 60                                                 SPECS     85
                50 CALL CARD(IAPTR,4HAXIS,.5,Y,IGN,-0)                     SPECS     86
                  C                                                        SPECS     87
                  C     IF PARAMETER CARDS EXIST, DISPLAY THE FIRST ONE.   SPECS     88
  105             C                                                        SPECS     89
                60 CALL DMGET(IPMPTXT,IPMPTR,IPPTR)                        SPECS     90
                  IF(IPPTR .EQ. 0) GO TO 80                                SPECS     91
                70 Y=Y-.5                                                  SPECS     92
                  CALL CARD(IPPTR,9HPARAMETER,.5,Y,ISIN,IPLTPTR)           SPECS     93
  110             C                                                        SPECS     94
                  C     CHECK FOR ADDITIONAL PARAMETER CARDS.              SPECS     95
                  C                                                        SPECS     96
                  CALL DMGET(IWCONE,IPPTR,IPPTR)                           SPECS     97
                  IF(IPPTR .EQ. 0) GO TO 80                                SPECS     98
```

```
      GO TO 70                                                          SPECS     99
C                                                                       SPECS    100
C         IF MULTIPLIER CARDS EXIST, DISPLAY THE FIRST ONE.             SPECS    101
C                                                                       SPECS    102
   80 CALL DMGET(IPMMTXT,IPMPTR,IMPTR)                                  SPECS    103
      IF(IMPTR .EQ. 0) GO TO 100                                        SPECS    104
   90 Y=Y-.5                                                            SPECS    105
      CALL CARD(IMPTR,10HMULTIPLIER,.5,Y,ISIN,IPLTPTR)                  SPECS    106
C                                                                       SPECS    107
C         CHECK FOR ADDITIONAL MULTIPLIER CARDS.                        SPECS    108
C                                                                       SPECS    109
      CALL DMGET(IWCONE,IMPTR,IMPTR)                                    SPECS    110
      IF(IMPTR .EQ. 0) GO TO 100                                        SPECS    111
      GO TO 90                                                          SPECS    112
C                                                                       SPECS    113
C         DETERMINE IF THERE IS ANOTHER PM GROUP IN THE AXIS GROUP.     SPECS    114
C                                                                       SPECS    115
  100 CALL DMGET(IPMNEXT,IPMPTR,IPMPTR)                                 SPECS    116
      IF(IPMPTR .EQ. 0) GO TO 110                                       SPECS    117
C                                                                       SPECS    118
C         THERE IS ANOTHER PM GROUP. DISPLAY "END" AND THE              SPECS    119
C         SPECIFICATIONS FOR THE NEXT PM GROUP.                         SPECS    120
      Y=Y-.5                                                            SPECS    121
      NN=NN+1                                                           SPECS    122
      CALL ANNOTA(.5,Y,3HEND,3,IDDAD(NN),ISIN,0,IPLTPTR,-0)             14APR76   27
C                                                                       SPECS    124
      CALL DMGET(IPMPLOT,IPMPTR,IPLTPTR)                                SPECS    125
      GO TO 60                                                          SPECS    126
C                                                                       SPECS    127
C         THE AXIS GROUP DOES NOT HAVE ANOTHER PM GROUP. DETERMINE      SPECS    128
C         WHETHER THE PLOT GROUP HAS ANOTHER AXIS GROUP.                SPECS    129
C                                                                       SPECS    130
  110 CALL DMGET(IAXNEXT,IAXPTR,IAXPTR)                                 SPECS    131
      IF(IAXPTR .EQ. 0) GO TO 120                                       SPECS    132
C                                                                       SPECS    133
C         THERE IS ANOTHER AXIS GROUP. DISPLAY IT.                      SPECS    134
C                                                                       SPECS    135
      CALL DMGET(IAXPM,IAXPTR,IPMPTR)                                   SPECS    136
      CALL DMGET(IPMPLOT,IPMPTR,IPLTPTR)                                SPECS    137
      GO TO 40                                                          SPECS    138
C                                                                       SPECS    139
C         THE PLOT GROUP DOES NOT HAVE ANOTHER AXIS GROUP.              SPECS    140
C         DETERMINE WHETHER THERE IS ANOTHER PLOT GROUP.                SPECS    141
C                                                                       SPECS    142
  120 CALL DMGET(IPGNEXT,IPGPTR,IPGHEAD)                                SPECS    143
C                                                                       SPECS    144
C         THERE IS ANOTHER PLOT GROUP.                                  SPECS    145
C                                                                       SPECS    146
      IF(IPGHEAD .NE. 0) GO TO 130                                      SPECS    147
C                                                                       SPECS    148
C         THERE ARE NO MORE PLOT GROUPS. DISPLAY "FINISH".              SPECS    149
C                                                                       SPECS    150
      Y=Y-.5                                                            SPECS    151
      NN=NN+1                                                           SPECS    152
      CALL ANNOTA(.5,Y,6HFINISH,6,IDDAD(NN),IGN,-0,-0,-0)               SPECS    153
  130 NMEND(2)=NN                                                       SPECS    154
      RETURN                                                            SPECS    155
```

```
      END                                                    SPECS     156
```

```
      OVERLAY(3,0)                                                      FRSTPLT    2
      PROGRAM FRSTPL2                                                   15FEB      1
C                                                                       FRSTPLT    4
C         THIS ROUTINE DISPLAYS THE PLOT SPECIFICATIONS, THE AXIS       FRSTPLT    5
C         SYSTEM, THE CURVE DESCRIPTIONS BOX, AND THE FIRST PLOT IN THE FRSTPLT    6
C         GROUP OF PLOTS TO BE PROCESSED.                               FRSTPLT    7
C                                                                       FRSTPLT    8
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1, /AXBEAD/  2
     X              IAXY2,IAXY3                                        /AXBEAD/   3
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/   2
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
      COMMON /GP/ IBUF(124),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY6777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,      /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/  3
     X               IPEMN,IPEMV                                        /PLTBEAD/  4
      COMMON /SAVCPPT/ NEXCPPT                                          /SAVCPPT/  2
C                                                                       FRSTPLT   14
C         DISPLAY THE PLOT SPECIFICATIONS.                              FRSTPLT   15
C                                                                       FRSTPLT   16
      CALL ORIGIN(-5.,8.5)                                              FRSTPLT   17
      CALL SQUARE(0.,-4.,10.,4.,IDSPEC(17),IGN,-0)                      FRSTPLT   18
      CALL DMGET(IPPG,IPLTPTR,IPGPTR)                                   FRSTPLT   19
      CALL DISCOND(IPGPTR,IPLTPTR)                                      5MAY76     8
      CALL DISMULT(IPLTPTR)                                             5MAY76     9
      CALL DMGET (IPNEXTC,IPLTPTR,NEXCPPT)                              14JUNE     9
C                                                                       FRSTPLT   22
C         CHECK FOR MULTIPLE Y-AXES. IF THEY EXIST, DISPLAY A FATAL     FRSTPLT   23
C         ERROR MESSAGE. THIS PROGRAM WILL NOT HANDLE MULTIPLE Y-AXES.  FRSTPLT   24
C                                                                       FRSTPLT   25
      CALL DMGET(IPAXIS,IPLTPTR,IAXPTR)                                 FRSTPLT   26
      CALL DMGET(IAXNYAX,IAXPTR,NYAXES)                                 FRSTPLT   27
      IF(NYAXES .NE. 1) CALL ERROR4(0.,-4.5,44HTHE PROGRAM WILL NOT HAND FRSTPLT   28
     XLE MULTIPLE Y-AXES.,44,7HFRSTPLT,5HFATAL)                         FRSTPLT   29
C                                                                       FRSTPLT   30
C         DISPLAY THE AXES, SCALES, AND TITLES.                         FRSTPLT   31
C                                                                       FRSTPLT   32
      CALL ORIGIN(-5.,-3.)                                              FRSTPLT   33
      CALL DISAXES(IPLTPTR)                                             FRSTPLT   34
      CALL LBLAXES(IPLTPTR)                                             FRSTPLT   35
      CALL DISTTL (IPLTPTR)                                             14JUNE    10
C                                                                       FRSTPLT   37
C         DISPLAY THE CURVE DESCRIPTIONS BOX.                           FRSTPLT   38
C                                                                       FRSTPLT   39
      CALL SQUARE (10.25,-2.,3.,10.,IDCD(21),IGN,-0)                    29APR      1
      CALL ANNOTA(10.95,9.35,5HCURVE,5,IDCD(22),IGN,-0,-0,-0)           FRSTPLT   41
      CALL ANNOTA(10.53,8.1,12HDESCRIPTIONS,12,IDCD(23),IGN,-0,-0,-0)   FRSTPLT   42
C                                                                       FRSTPLT   43
C         DISPLAY THE PLOT.                                             FRSTPLT   44
C                                                                       FRSTPLT   45
      CALL DISPLOT(IPLTPTR)                                             FRSTPLT   46
C                                                                       FRSTPLT   47
C         PASS "IPLTPTR" THROUGH COMMON/BEADS/ TO TASK QTRBACK.         FRSTPLT   48
C                                                                       FRSTPLT   49
      CALL RETSKD(7HQTRBACK)                                            FRSTPLT   50
```

```
      OVERLAY(3,0)                                                      FRSTPLT    2
      PROGRAM FRSTPL2                                                   15FEB      1
C                                                                       FRSTPLT    4
C         THIS ROUTINE DISPLAYS THE PLOT SPECIFICATIONS, THE AXIS       FRSTPLT    5
C         SYSTEM, THE CURVE DESCRIPTIONS BOX, AND THE FIRST PLOT IN THE FRSTPLT    6
C         GROUP OF PLOTS TO BE PROCESSED.                               FRSTPLT    7
C                                                                       FRSTPLT    8
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1, /AXBEAD/  2
     X              IAXY2,IAXY3                                        /AXBEAD/   3
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/   2
      COMMON/DISPLAY/IDSTOP,IDOAD(90),NN,NNEND(10),IOSPEC(19),IOCF(41), /DISPLAY/  2
     X               IOAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IOINSTR(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT      1
      COMMON /GP/ IBUF(124),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY6777     1
     X     ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,      /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/  3
     X               IPEMN,IPEMV                                        /PLTBEAD/  4
      COMMON /SAVCPPT/ NEXCPPT                                          /SAVCPPT/  2
C                                                                       FRSTPLT   14
C         DISPLAY THE PLOT SPECIFICATIONS.                              FRSTPLT   15
C                                                                       FRSTPLT   16
      CALL ORIGIN(-5.,8.5)                                              FRSTPLT   17
      CALL SQUARE(0.,-4.,10.,4.,IOSPEC(17),IGN,-0)                      FRSTPLT   18
      CALL CMGET(IPPG,IPLTPTR,IPGPTR)                                   FRSTPLT   19
      CALL DISCOND(IPGPTR,IPLTPTR)                                      5MAY76     8
      CALL DISMULT(IPLTPTR)                                             5MAY76     9
      CALL CMGET (IPNEXTC,IPLTPTR,NEXCPPT)                              14JUNE     9
C                                                                       FRSTPLT   22
C         CHECK FOR MULTIPLE Y-AXES. IF THEY EXIST, DISPLAY A FATAL     FRSTPLT   23
C         ERROR MESSAGE. THIS PROGRAM WILL NOT HANDLE MULTIPLE Y-AXES.  FRSTPLT   24
C                                                                       FRSTPLT   25
      CALL CMGET(IPAXIS,IPLTPTR,IAXPTR)                                 FRSTPLT   26
      CALL CMGET(IAXNYAX,IAXPTR,NYAXES)                                 FRSTPLT   27
      IF(NYAXES .NE. 1) CALL ERROR4(0.,-4.5,44HTHE PROGRAM WILL NOT HANG FRSTPLT   28
     XLE MULTIPLE Y-AXES.,44,7HFRSTPLT,5HFATAL)                         FRSTPLT   29
C                                                                       FRSTPLT   30
C         DISPLAY THE AXES, SCALES, AND TITLES.                         FRSTPLT   31
C                                                                       FRSTPLT   32
      CALL ORIGIN(-5.,-3.)                                              FRSTPLT   33
      CALL DISAXES(IPLTPTR)                                             FRSTPLT   34
      CALL LBLAXES(IPLTPTR)                                             FRSTPLT   35
      CALL DISTTL (IPLTPTR)                                             14JUNE    10
C                                                                       FRSTPLT   37
C         DISPLAY THE CURVE DESCRIPTIONS BOX.                           FRSTPLT   38
C                                                                       FRSTPLT   39
      CALL SQUARE (10.25,-2.,3.,11.,IOCD(21),IGN,-0)                    29APR      1
      CALL ANNOTA(10.95,9.35,5HCURVE,5,IOCD(22),IGN,-0,-0,-0)           FRSTPLT   41
      CALL ANNOTA(10.53,8.1,12HDESCRIPTIONS,12,IOCD(23),IGN,-0,-0,-0)   FRSTPLT   42
C                                                                       FRSTPLT   43
C         DISPLAY THE PLOT.                                             FRSTPLT   44
C                                                                       FRSTPLT   45
      CALL DISPLOT(IPLTPTR)                                             FRSTPLT   46
C                                                                       FRSTPLT   47
C         PASS "IPLTPTR" THROUGH COMMON/BEADS/ TO TASK QTRBACK.         FRSTPLT   48
C                                                                       FRSTPLT   49
      CALL RETSYC(7HQTRBACK)                                            FRSTPLT   50
```

```
      END                                                    FRSTPLT   51
```

```
      SUBROUTINE DCRVDES(ICRVPTR,ISYMBOL,NCURVE)                        DCRVDES    2
C                                                                       DCRVDES    3
C         THIS ROUTINE BUILDS A CURVE DESCRIPTION AND DISPLAYS IT.      DCRVDES    4
C              ICRVPTR - THE CURVE BEAD POINTER.                        DCRVDES    5
C              ISYMBOL - THE CURVE PLOT SYMBOL.                         DCRVDES    6
C               NCURVE - THE NUMBER OF THE CURVE. THE CURVES ARE        DCRVDES    7
C                        NUMBERED CHRONOLOGICALLY AS THEY APPEAR        DCRVDES    8
C                        ON THE SCREEN. THAT NUMBER IS USED IN          DCRVDES    9
C                        THIS SUBROUTINE TO DETERMINE THE POSITION      DCRVDES   10
C                        ON THE SCREEN WHERE THE CURVE DESCRIPTION      DCRVDES   11
C                        WILL BE DISPLAYED.                             DCRVDES   12
C                                                                       DCRVDES   13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNENO(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
     Y               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYB777     1
     Y     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      INTEGER RECORD(3)                                                 DCRVDES   17
C                                                                       DCRVDES   18
C         GET THE PARAMETER NAME AND VALUE FROM THE CURVE BEAD, ENCODE  DCRVDES   19
C         THE CURVE DESCRIPTION, AND DISPLAY IT.                        DCRVDES   20
C                                                                       DCRVDES   21
      CALL DMGET(ICPARAN,ICRVPTR,NAME)                                  DCRVDES   22
      CALL DMGET(ICPARAV,ICRVPTR,VALUE)                                 DCRVDES   23
      IDEC=0                                                            FIXINH
      DO 10 J=1,9                                                       FIXINH
      IF (VALUE/10.**(J-1).LE.1.) GO TO 20                              FIXINH     4
   10 IDEC=IDEC-1                                                       FIXINH     5
      IDEC=0                                                            FIXINH     6
   20 ENCODE (25,1000,RECORD) ISYMBOL,NAME,IDEC,VALUE                   FIXINH     7
 1000 FORMAT(A1,1X,A10,1X,F12.*)                                        FIXINH     8
      Y = 8.0 - .5 * FLOAT(NCURVE)                                      DCRVDES   26
      CALL ANNOTA (10.43,Y,RECORD,25,IDCO(NCURVE),IGN,NCURVE,-0,-0)     6JULY77    6
      RETURN                                                            DCRVDES   28
      END                                                               DCRVDES   29
```

```
      SUBROUTINE DISAXES(IPLTPTR)                                        DISAXES    2
C                                                                        DISAXES    3
C     THIS ROUTINE:                                                      DISAXES    4
C          1 - RETRIEVES AND COMPUTES ALL PLOT SCALING INFORMATION.      DISAXES    5
C          2 - DISPLAYS A NINE INCH X-AXIS, A SIX INCH Y-AXIS, AND       DISAXES    6
C              SCALE INFORMATION FOR BOTH.                               DISAXES    7
C                                                                        DISAXES    8
C     PRIMARY INPUT TO THE ROUTINE IS A POINTER TO THE PLOT BEAD OF      DISAXES    9
C     THE PLOT FOR WHICH AXES ARE TO BE DRAWN. THE AXES SCALE DATA       DISAXES   10
C     IS OUTPUT IN COMMON/AXSCALE/.                                      DISAXES   11
C                                                                        DISAXES   12
      COMMON/AXSBEAD/IAXSXM,IAXSXI,IAXSY1M,IAXSY1I,IAXSY2M,IAXSY2I,      /AXSBEAD/   2
     X               IAXSY3M,IAXSY3I                                     /AXSBEAD/   3
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/   2
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IOSPEC(19),IOCF(41),  /DISPLAY/   2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),          6JULY77     2
     X               IOINSTR(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT       1
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/   3
     X               IPFMN,IPFMV                                        /PLTBEAD/   4
C                                                                        DISAXES   17
C     GET THE MINIMUMS AND INCREMENTS FROM THE AXIS                      DISAXES   18
C     SCALE BEAD AND COMPUTE THE MAXIMUMS.                               DISAXES   19
C                                                                        DISAXES   20
      CALL DMGET(IPAXISS,IPLTPTR,IAXSPTR)                                DISAXES   21
      CALL DMGET(IAXSXM,IAXSPTR,XMIN)                                    DISAXES   22
      CALL DMGET(IAXSXI,IAXSPTR,XINCR)                                   DISAXES   23
      CALL DMGET(IAXSY1M,IAXSPTR,YMIN)                                   DISAXES   24
      CALL DMGET(IAXSY1I,IAXSPTR,YINCR)                                  DISAXES   25
      XMAX = XMIN + 9. * XINCR                                           DISAXES   26
      YMAX = YMIN + 6. * YINCR                                           DISAXES   27
C                                                                        DISAXES   28
C     IF INCR IS NEGATIVE, THEN MIN IS IN FACT THE MAXIMUM AND MAX       DISAXES   29
C     IS IN FACT THE MINIMUM. THE FOLLOWING CODE INSURES THAT MIN        DISAXES   30
C     AND MAX ARE IN FACT THE MINIMUM AND MAXIMUM RESPECTIVELY.          DISAXES   31
C                                                                        DISAXES   32
      IF(XMIN .LE. XMAX) GO TO 10                                        DISAXES   33
      TEMP=XMIN                                                          DISAXES   34
      XMIN=XMAX                                                          DISAXES   35
      XMAX=TEMP                                                          DISAXES   36
C                                                                        DISAXES   37
   10 IF(YMIN .LE. YMAX) GO TO 20                                        DISAXES   38
      TEMP = YMIN                                                        DISAXES   39
      YMIN = YMAX                                                        DISAXES   40
      YMAX = TEMP                                                        DISAXES   41
C                                                                        DISAXES   42
   20 XINCR=ABS(XINCR)                                                   DISAXES   43
      YINCR = ABS(YINCR)                                                 DISAXES   44
C                                                                        DISAXES   45
C     DISPLAY THE AXES.                                                  DISAXES   46
C                                                                        DISAXES   47
      CALL DMSET (IAXSXM,IAXSPTR,XMIN)                                   20SEPT      1
      CALL DMSET (IAXSXI,IAXSPTR,XINCR)                                  20SEPT      2
      CALL DMSET (IAXSY1M,IAXSPTR,YMIN)                                  20SEPT      3
      CALL DMSET (IAXSY1I,IAXSPTR,YINCR)                                 20SEPT      4
      CALL GAXIS(0.,0.,9.,0.,XMIN,XINCR,-1,IOAXIS(1),-0,-0)              DISAXES   48
      CALL GAXIS(0.,0.,6.,90.,YMIN,YINCR,1,IOAXIS(2),-0,-0)              DISAXES   49
```

```
      RETURN                                                    DISAXES   50
      END                                                       DISAXES   51
```

```
      SUBROUTINE DISCOND(IPGPTR,IPLTPTR)                                5MAY76  10
C                                                                       DISCOND   3
C         THIS ROUTINE DISPLAYS THE CURRENT CONDITIONS.                 DISCOND   4
C                                                                       DISCOND   5
C         WORD IDSPEC(18) IS A STATUS WORD FOR THE CONDITIONS. IF IT IS DISCOND   9
C         ZERO, THEN NO CONDITIONS ARE DISPLAYED. IF IT IS NOT ZERO,    DISCOND  10
C         THEN CONDITIONS ARE DISPLAYED AND IDSPEC(18) CAN BE USED TO   DISCOND  11
C         ERASE THEM. THE DISPLAY ADDRESSES OF THE CONDITIONS ARE STORED DISCOND  12
C         IN IDSPEC(1) THROUGH IDSPEC(IDSPEC(18)).                      DISCOND  13
C                                                                       DISCOND  14
      COMMON/COMPCOD/IWDZERO,IWDONE,IWDTWO,IWDTHRE,IWDFOUR,IWDFIVE,     /COMPCOD/  2
     X               IWDSIX,IWDSEVE,IWDEIGH,IWDNINE,IWDTEN             /COMPCOD/  3
      COMMON/DISPLAY/IDSTOF,IDDAD(81),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77   2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBER77   1
     X     ISTR,IBUT,IMASK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77   3
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,    /PGBEAD/   2
     X              IPGLOA                                             /PGBEAD/   3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,      /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/   3
     X               IPFMN,IPFMV                                        /PLTBEAD/   4
      INTEGER STRING(8)                                                 DISCOND  20
      LINECNT=0                                                         DISCOND  21
C                                                                       DISCOND  22
C         GET THE NUMBER OF LINES OF CONDITION SPECIFICATIONS FROM THE  DISCOND  23
C         PLOT GROUP BEAD. BECAUSE OF CONTINUATION CARDS THE NUMBER OF  DISCOND  24
C         LINES IS NOT NECESSARILY THE SAME AS THE NUMBER OF CONDITIONS. DISCOND  25
C                                                                       DISCOND  26
      CALL DMGET(IPGNCNS,IPGPTR,NCOND)                                  DISCOND  27
      IF(NCOND .GT. 9) GO TO 10                                         DISCOND  28
C                                                                       DISCOND  29
C         THERE ARE NO CONDITIONS.                                      DISCOND  30
C                                                                       DISCOND  31
      IDSPEC(18) = 0                                                    DISCOND  32
      RETURN                                                            DISCOND  33
C                                                                       DISCOND  34
C         GET THE NUMBER OF MULTIPLIERS FROM THE PLOT BEAD.             DISCOND  35
C                                                                       DISCOND  36
   10 CALL IMGET(IPNMULT,IPLTPTR,NMULT)                                 DISCOND  37
      IF((NCOND + NMULT + 2) .LE. 16) GO TO 20                          DISCOND  38
C                                                                       DISCOND  39
C         THE TOTAL NUMBER OF LINES REQUIRED TO DISPLAY THE PLOT SPECS  DISCOND  40
C         EXCEEDS THE NUMBER OF LINES ALLOCATED FOR THEIR DISPLAY. NO   DISCOND  41
C         PLOT SPECIFICATIONS WILL BE DISPLAYED.                        DISCOND  42
C                                                                       DISCOND  43
      IDSPEC(18) = 0                                                    DISCOND  44
      CALL ERROFM(0.,-4.5,70HTHE PLOT SPECS EXCEED THE ALLOCATED SPACE. DISCOND  45
     XNO SPECS WILL BE DISPLAYED.,70,7HDISCOND,6HINFORM)                DISCOND  46
      RETURN                                                            DISCOND  47
C                                                                       DISCOND  48
C         DISPLAY THE CONDITION(S) HEADING.                             DISCOND  49
C                                                                       DISCOND  50
   20 LINECNT = LINECNT + 1                                             DISCOND  51
      Y=(-.25)*FLOAT(LINECNT)                                           DISCOND  52
      CALL ANNOTA(.2,Y,12HCONDITION(S),12,IDSPEC(LINECNT),IGN,-0,-0,-0) DISCOND  53
C                                                                       DISCOND  54
```

210

```
      C          LOOP THROUGH THE CONDITION TEXT BEADS.                    DISCOND   55
      C                                                                    DISCOND   56
60              CALL DMGET(IPGCTXT,IPGPTR,ICTPTR)                          DISCOND   57
                DO 30 I=1,NCOND                                            DISCOND   58
      C                                                                    DISCOND   59
      C          GET THE CONDITION TEXT FROM THE CONDITION                 DISCOND   60
      C          TEXT BEAD AND DISPLAY IT.                                 DISCOND   61
65    C                                                                    DISCOND   62
                CALL DMGET(INCTWO,ICTPTR,NCHAR)                            DISCOND   63
                CALL DMGET(7H001C0000036,ICTPTR,STRING)                    DISCOND   64
                LINECNT = LINECNT + 1                                      DISCOND   65
                Y=(-.25)*FLOAT(LINECNT)                                    DISCOND   66
70              CALL ANNOTA(.2,Y,STRING,NCHAR,IOSPEC(LINECNT),IGN,-0,-0,-0) DISCOND  67
      C                                                                    DISCOND   68
      C          GET THE POINTER TO THE NEXT CONDITION TEXT BEAD.          DISCOND   69
      C          (THE LAST EXECUTION OF THE NEXT STATEMENT IS SUPERFLOUS.) DISCOND   70
      C                                                                    DISCOND   71
75              CALL DMGET(INCONE,ICTPTR,ICTPTR)                           DISCOND   72
             30 CONTINUE                                                   DISCOND   73
                IOSPEC(18) = LINECNT                                       DISCOND   74
                RETURN                                                     DISCOND   75
                END                                                        DISCOND   76
```

```
      SUBROUTINE DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NPLOT)                 ADPTPRE    1
C                                                                               DISCRVE    3
C         THIS ROUTINE DISPLAYS THE CURVE(SET OF POINTS) INDICATED BY           DISCRVE    4
C         "ICRVPTR". THE PRIMARY INPUTS TO THE ROUTINE ARE THE POINTER          DISCRVE    5
C         TO THE CURVE BEAD AND THE PLOT SCALE DATA. PRIMARY OUTPUTS            DISCRVE    6
C         FROM THE ROUTINE ARE:                                                 DISCRVE    7
C              ISYMBOL - THE SYMBOL USED TO DISTINGUISH THIS CURVE FROM         DISCRVE    8
C                        OTHER CURVES ON THE PLOT.                              DISCRVE    9
C               INITSS - THE DISPLAY ADDRESSES FOR THE CURVE ARE STORED         DISCRVE   10
C                        IN ARRAY IDDAD. IDDAD(INITSS) CONTAINS THE             DISCRVE   11
C                        DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED          DISCRVE   12
C                        TO EXHIBIT THE CURVE.(INITSS-INITIAL SUBSCRIPT)        DISCRVE   13
C              NUMDISP - NUMDISP CONTAINS THE NUMBER OF DISPLAYS                DISCRVE   14
C                        REQUIRED TO EXHIBIT THE CURVE.                         DISCRVE   15
C              NPLOT   - NPLOT CONTAINS THE NUMBER OF THE                       ADPTPRE    2
C                        PLOT IN THE TABLE                                      ADPTPRE    3
C         INITSS AND NUMDISP ARE NEEDED TO ERASE THE CURVE. THEY WILL BE        DISCRVE   16
C         STORED IN THE CURVE DISPLAY ACCESS TABLE.                             DISCRVE   17
C                                                                               DISCRVE   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                            /AXSCALE/  2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,          /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                         /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),         /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),                  6JULY77    2
     X               IDINSTR(19),ICDATA3(20,4,2),NCIP,IDBLOCK(20),NCIPN         1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                          CY8777     1
     X     ISTR,IBUT,IPARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                         6JULY77    3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                        /PARMSV/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                     /POINTS/   2
      COMMON /SYMTAB/ ISYMTAB(20)                                               /SYMTAB/   2
      DIMENSION MESG(7)                                                         9JULY76    1
C                                                                               DISCRVE   28
C         RECORD THE INITIAL VALUE OF THE IDDAD SUBSCRIPT.                      DISCRVE   29
C                                                                               DISCRVE   30
      INITSS=(NPLOT-1)*4+1                                                      ADPTPRE    4
      NN=INITSS-1                                                               ADPTPRE    5
C                                                                               DISCRVE   32
C         GET THE NUMBER OF POINTS ON THE CURVE, THE CURVE SYMBOL               DISCRVE   33
C         NUMBER, THE RECORD BEAD POINTER, THE X-COORDINATE BEAD                DISCRVE   34
C         POINTER, AND THE Y-COORDINATE BEAD POINTER FROM THE                   DISCRVE   35
C         CURVE BEAD.                                                           DISCRVE   36
C                                                                               DISCRVE   37
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                        DISCRVE   38
      CALL DMGET(ICSYM,ICRVPTR,ISYMBOL)                                         DISCRVE   39
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                         DISCRVE   40
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                          DISCRVE   41
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                          DISCRVE   42
C                                                                               DISCRVE   43
C         IF THE NUMBER OF POINTS EXCEEDS THE PROGRAMMED LIMIT(248),            DISCRVE   44
C         DISPLAY THE FIRST 248 POINTS ONLY.                                    DISCRVE   45
C                                                                               DISCRVE   46
      IF(NPOINTS .LE. 248) GO TO 11                                             DISCRVE   47
      NPOINTS=248                                                               DISCRVE   48
      CALL DMSET(ICNPTS,ICRVPTR,NPOINTS)                                        DISCRVE   49
      CALL ERRPRM(0.,7.,42HPOINTS LIMIT EXCEEDED. THE FIRST 248 POINTS W        DISCRVE   50
     XILL BE DISPLAYED.,62,7HDISCRVE,6HINFORM)                                  DISCRVE   51
C                                                                               DISCRVE   52
```

```
      C         CONVERT THE CURVE SYMBOL NUMBER INTO A CURVE SYMBOL.                DISCRVE    53
      C                                                                             DISCRVE    54
   60    10 ISYMBOL = ISYMTAB(ISYMBOL)                                              DISCRVE    55
      C                                                                             DISCRVE    56
      C         MOVE THE CONTENTS OF THE RECORD NUMBER BEAD, THE X-COORDINATE       DISCRVE    57
      C         BEAD AND THE Y-COORDINATE BEAD INTO CORE. THE RECORD NUMBER         DISCRVE    58
      C         BEAD IS A STATUS BEAD FOR THE POINTS IN THE OTHER TWO BEADS.        DISCRVE    59
   65 C              ACTIVE - THE RECORD NUMBER IS POSITIVE OR SHADDED. THE         DISCRVE    60
      C                       RECORD NUMBER IS SHADDED IF AND ONLY IF THE           DISCRVE    61
      C                       POINT HAS BEEN ADDED DURING AN EDIT.                  DISCRVE    62
      C            INACTIVE - THE RECORD NUMBER IS NEGATIVE. A POINT IS           DISCRVE    63
      C                       INACTIVE IF AND ONLY IF IT HAS BEEN DELETED          DISCRVE    64
   70 C                       DURING AN EDIT.                                       DISCRVE    65
      C                                                                             DISCRVE    66
         KOMPCOD = 700000000018 + NPOINTS * 8**6                                    DISCRVE    67
         CALL GMGET(KOMPCOD,IRECPTR,NREC)                                           DISCRVE    68
         CALL GMGET(KOMPCOD,IXPTR,X)                                                DISCRVE    69
   75    CALL GMGET(KOMPCOD,IYPTR,Y)                                                DISCRVE    70
      C                                                                             DISCRVE    71
      C         LOOP THROUGH THE POINTS. IF A POINT IS INACTIVE OR NOT ON THE       DISCRVE    72
      C         PLOT, SKIP IT. OTHERWISE, CONVERT THE COORDINATES TO DGUS AND       DISCRVE    73
      C         PACK THE DESCRIPTION OF THE POINT INTO THE DESCRIPTION BUFFER.      DISCRVE    74
   80 C                                                                             DISCRVE    75
         12 DO 20 I=1,NPOINTS                                                       9JUL76     2
            IF(NREC(I) .LT. 0) GO TO 20                                             DISCRVE    77
            IF ((X(I).LT.XMIN).OR.(X(I).GT.XMAX))GO TO 20                           30JULY76   1
            IF ((Y(I).LT.YMIN).OR.(Y(I).GT.YMAX))GO TO 20                           30JULY76   2
   85       IX=IFIX(((X(I)-XMIN)*DGUPIN)/XINCR+XORDGU-6)                            50CT       1
            IY=IFIX(((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU - 11.5)                 DISCRVE    81
            CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                               DISCRVE    82
            CALL GUAN(ISYMBOL,1,IBUF,NBYTE,MBYTE)                                   DISCRVE    83
            IF(NBYTE .LT. 312) GO TO 20                                             DISCRVE    84
   90 C                                                                             DISCRVE    85
      C         THE DESCRIPTION BUFFER IS FULL. FLUSH IT.                           DISCRVE    86
      C                                                                             DISCRVE    87
            NN=NN+1                                                                 DISCRVE    88
            CALL GIDISP(NCON,IBUF,NBYTE,IODAD(NN),IGN)                              DISCRVE    89
   95    20 CONTINUE                                                               DISCRVE    90
      C                                                                             DISCRVE    91
      C         ALL POINTS HAVE BEEN CONSIDERED. IS THE DESCRIPTION                 DISCRVE    92
      C         BUFFER EMPTY? IF NOT, FLUSH IT.                                     DISCRVE    93
      C                                                                             DISCRVE    94
  100       IF(NBYTE .EQ. 0) GO TO 30                                             DISCRVE    95
            NN=NN+1                                                                 DISCRVE    96
            CALL GIDISP(NCON,IBUF,NBYTE,IODAD(NN),IGN)                              DISCRVE    97
         30 NUMDISP = NN - INITSS + 1                                               DISCRVE    98
            RETURN                                                                  DISCRVE    99
  105       END                                                                     DISCRVE   100
```

```
      SUBROUTINE DISMULT(IPLTPTR)                                        5MAY76     1
C                                                                        DISMULT    3
C         THIS ROUTINE DISPLAYS THE MULTIPLIERS AT THE TOP OF THE PLOT.  DISMULT    4
C                                                                        DISMULT    5
C         WORD IDSPEC(19) IS A STATUS WORD FOR THE MULTIPLIERS. IF IT IS DISMULT    6
C         ZERO, THEN NO MULTIPLIERS ARE DISPLAYED. IF IT IS NOT ZERO,    DISMULT    7
C         THEN MULTIPLIERS ARE DISPLAYED AND IDSPEC(19) CAN BE USED IN   DISMULT    8
C         CONJUNCTION WITH IDSPEC(18) TO ERASE THEM. THE DISPLAY         DISMULT    9
C         ADDRESSES OF THE MULTIPLIERS ARE STORED IN                     DISMULT   10
C         IDSPEC(IDSPEC(18) + 1) THROUGH IDSPEC(IDSPEC(19)).             DISMULT   11
C                                                                        DISMULT   12
      COMMON/DISPLAY/IDSTOP,IODAD(82),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),IODATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYBTTT     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77    3
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,     /PGBEAD/    2
     X              IPGLCA                                              /PGBEAD/    3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/    2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,        /PLTBEAD/    3
     Y               IPFMN,IPFMV                                         /PLTBEAD/    4
      INTEGER RECORD(2)                                                  DISMULT   17
C                                                                        DISMULT   18
C         GET THE NUMBER OF MULTIPLIERS FROM THE PLOT BEAD.              DISMULT   19
C         THE NUMBER OF LINES OF CONDITION SPECIFICATIONS                5MAY76     2
C         WHICH HAVE BEEN DISPLAYED IS IN IDSPEC(18). CHECK              5MAY76     3
C         TO SEE IF THE MULTIPLIERS CAN BE DISPLAYED.                    5MAY76
C                                                                        DISMULT
      CALL DMGET(IPNMULT,IPLTPTR,NMULT)                                  DISMULT   22
      IF(NMULT .GT. 0) GO TO 10                                          DISMULT   22
      IDSPEC(19) = 0                                                     DISMULT   23
      RETURN                                                             DISMULT   24
   10 LINECNT = IDSPEC(18)                                               5MAY76     5
      IF((LINECNT + NMULT +1) .LE. 16) GO TO 20                          5MAY76     6
      IDSPEC(19) = 0                                                     DISMULT   30
      RETURN                                                             DISMULT   31
C                                                                        DISMULT   32
C         DISPLAY THE MULTIPLIER(S) HEADING.                             DISMULT   33
C                                                                        DISMULT   34
   20 LINECNT = LINECNT + 1                                              DISMULT   35
      Y = (-.25) * FLOAT(LINECNT)                                        DISMULT   36
      CALL ANNOTA(.2,Y,13HMULTIPLIER(S),13,IDSPEC(LINECNT),IGN,-0,-0,-0) DISMULT   37
C                                                                        DISMULT   38
C         LOOP THROUGH THE MULTIPLIERS.                                  DISMULT   39
C                                                                        DISMULT   40
      DO 30 I=1,NMULT                                                    DISMULT   41
C                                                                        DISMULT   42
C         GET THE MULTIPLIER NAME AND VALUE FROM THE                     DISMULT   43
C         PLOT BEAD, ENCODE IT, AND DISPLAY IT.                          DISMULT   44
C                                                                        DISMULT   45
      CALL DMGET(IPFMN+2*I-2,IPLTPTR,NAME)                               DISMULT   46
      CALL DMGET(IPFMV+2*I-2,IPLTPTR,VALUE)                              DISMULT   47
      ENCODE(20,1000,RECORD) NAME,VALUE                                  DISMULT   48
 1000 FORMAT(A7,"=",G12.6)                                               DISMULT   49
      LINECNT = LINECNT + 1                                              DISMULT   50
      Y = (-.25) * FLOAT(LINECNT)                                        DISMULT   51
      CALL ANNOTA(.2,Y,RECORD,20,IDSPEC(LINECNT),IGN,-0,-0,-0)           DISMULT   52
```

```
      30 CONTINUE                                                    DISMULT    53
         IDSPEC(19) = LINECNT                                        DISMULT    54
60       RETURN                                                      DISMULT    55
         END                                                         DISMULT    56
```

```
      SUBROUTINE DISPLOT(IPLTPTR)                                        DISPLOT    2
C                                                                        DISPLOT    3
C         THIS ROUTINE DISPLAYS ALL OF THE ACTIVE CURVES IN A PLOT.      DISPLOT    4
C         THE PRIMARY INPUT TO THE ROUTINE IS THE PLOT BEAD POINTER.     DISPLOT    5
C                                                                        DISPLOT    6
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,   /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                  /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),           6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT     1
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/  2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,        /PLTBEAD/  3
     X               IPFMN,IPFMV                                         /PLTBEAD/  4
      CALL ORIGIN(-5.,-3.)                                               DISPLOT   10
C                                                                        DISPLOT   11
C         INITIALIZE NCIP. NCIP IS THE NUMBER OF CURVES IN A PLOT. MORE  DISPLOT   12
C       - SPECIFICALLY, NCIP IS THE NUMBER OF CURVES WHOSE ACCESS DATA   DISPLOT   13
C         IS STORED IN THE CURVE DISPLAY ACCESS TABLE(SEE SUBROUTINE     DISPLOT   14
C         ICDATAB FOR AN EXPLANATION OF THE CURVE DISPLAY ACCESS TABLE). DISPLOT   15
C         THE MAXIMUM VALUE OF NCIP IS TWENTY, I.E., THERE CAN BE NO     DISPLOT   16
C         MORE THAN TWENTY CURVES PER PLOT.                              DISPLOT   17
C                                                                        DISPLOT   18
      NCIP=0                                                             DISPLOT   19
      DO 5 J=1,41                                                        2NOV       1
    5 IDCF(J)=0                                                          2NOV       2
C                                                                        DISPLOT   20
C         GET THE POINTER TO THE FIRST CURVE BEAD FROM THE PLOT BEAD.    DISPLOT   21
C                                                                        DISPLOT   22
      CALL DMGET(IPCURVE,IPLTPTR,ICRVPTR)                                DISPLOT   23
      GO TO 20                                                           DISPLOT   24
C                                                                        DISPLOT   25
C         GET THE POINTER TO THE NEXT CURVE BEAD.                        DISPLOT   26
C                                                                        DISPLOT   27
   10 CALL DMGET(ICNEXT,ICRVPTR,ICRVPTR)                                 DISPLOT   28
      IF(ICRVPTR .EQ. 0) RETURN                                          DISPLOT   29
C                                                                        DISPLOT   30
C         A CURVE BEAD EXISTS. IS THE CURVE ACTIVE OR INACTIVE?          DISPLOT   31
C              ACTIVE - ISTAT = 0                                        DISPLOT   32
C            INACTIVE - ISTAT = 8HINACTIVE                               14APR76    2
C         A CURVE IS INACTIVE IF AND ONLY IF IT HAS BEEN DELETED DURING  DISPLOT   34
C         AN EDIT.                                                       DISPLOT   35
C                                                                        DISPLOT   36
   20 CALL DMGET(ICSTAT,ICRVPTR,ISTAT)                                   DISPLOT   37
      IF(ISTAT .EQ. 8HINACTIVE) GO TO 10                                 14APR76    3
C                                                                        DISPLOT   39
C         THE CURVE IS ACTIVE. HAS THE CURVE LIMIT BEEN REACHED?         DISPLOT   40
C                                                                        DISPLOT   41
      IF(NCIP .LT. 20) GO TO 30                                          DISPLOT   42
C                                                                        DISPLOT   43
C         THE CURVE LIMIT HAS REACHED ON THE PREVIOUS CURVE.             DISPLOT   44
C                                                                        DISPLOT   45
      CALL ERRORM(0.,7.,72HTHERE ARE MORE THAN TWENTY CURVES IN THE PLOT DISPLOT   46
     X. TWENTY WILL BE DISPLAYED.,72,7HDISPLOT,6HINFORM)                 DISPLOT   47
      RETURN                                                             DISPLOT   48
C                                                                        DISPLOT   49
C         THE CURVE LIMIT HAS NOT BEEN REACHED. DISPLAY THE CURVE AND    DISPLOT   50
C         ITS DESCRIPTION AND UPDATE THE CURVE DISPLAY ACCESS TABLE.     DISPLOT   51
```

```
      C                                                                 DISPLOT    52
         30 CALL DISCRVE (ICRVPTR,ISYM80L,INITSS,NUMDISP,NCIP+1)        ADPTPRE     6
60          IF(NUMDISP .EQ. 0) GO TO 10                                DISPLOT    54
            NCIP=NCIP+1                                                DISPLOT    55
            NCIPP=NCIP                                                 1SEPT      2
            CALL DCRVDES(ICRVPTR,ISYM80L,NCIP)                         DISPLOT    56
            CALL UCDATA8(NCIP,1,6HACTIVE,INITSS,NUMDISP,ICRVPTR)       14APR76    4
65          CALL CMGET (ICFIT,ICRVPTR,IFT)                             13SEPT     2
            IF (IFT.EQ.0) CALL UCDATA8 (NCIP,2,8HINACTIVE,0,0,0)       13SEPT     3
            IF (IFT.GT.0) CALL REPLCF (NCIP,ICRVPTR,IFT,0)             20SEPT     5
            GO TO 10                                                   DISPLOT    58
            END                                                        DISPLOT    59
```

```
      SUBROUTINE DISTITL(IPLTPTR)                                        14JUNE    11
C                                                                        DISTITL    3
C         THIS ROUTINE DISPLAYS THE TITLE AT THE TOP OF THE              DISTITL    4
C         PLOT. THE NUMBER OF LINES IS LIMITED TO THREE.                 DISTITL    5
C                                                                        DISTITL    6
      COMMON/COMPCOD/IWDZERO,IWDONE,IWDTWO,IWDTHRE,IWDFOUR,IWDFIVE,      /COMPCOD/   2
     X               IWDSIX,IWDSEVE,IWDEIGH,IWDNINE,IWDTEN               /COMPCOD/   3
      COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNEND(10),IOSPEC(19),IDCF(41),  /DISPLAY/   2
     X               IOAXIS(4),IDTITLE(4),IDCD(23),IOMENU(20),          6JULY77     2
     X               IOINSTR(19),ICOATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT       1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYBER77     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77     3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/   3
     X               IPFMN,IPFMV                                        /PLTBEAD/   4
      INTEGER STRING(8)                                                  DISTITL    11
      CALL ORIGIN (-5.,-7.)                                              14JUNE     13
      Y=6.5                                                              DISTITL    12
      LINECNT=0                                                          DISTITL    13
C                                                                        DISTITL    14
C         GET THE TITLE TEXT POINTER FROM THE PLOT GROUP BEAD. IF THE    DISTITL    15
C         POINTER IS ZERO, THEN THERE IS NO TITLE TO DISPLAY.            DISTITL    16
C                                                                        DISTITL    17
      CALL DMGET (IPTITLE,IPLTPTR,ITPTR)                                 14JUNE     14
   10 IF(ITPTR .EQ. 0) GO TO 20                                          DISTITL    19
C                                                                        DISTITL    20
C         THERE IS A LINE OF TITLE TO BE DISPLAYED. GET THE NUMBER       DISTITL    21
C         OF CHARACTERS AND THE TITLE TEXT FROM THE TITLE TEXT BEAD      DISTITL    22
C         AND DISPLAY THE LINE.                                          DISTITL    23
C                                                                        DISTITL    24
      CALL DMGET(IWDTWO,ITPTR,NCHAR)                                     DISTITL    25
      CALL DMGET(7001000000039,ITPTR,STRING)                             DISTITL    26
      Y=Y-.25                                                            DISTITL    27
      LINECNT = LINECNT + 1                                              DISTITL    28
      CALL ANNOTA(.42,Y,STRING,NCHAR,IDTITLE(LINECNT),IGN,-0,-0,-0)      DISTITL    29
C                                                                        DISTITL    30
C         IF THE LINE LIMIT HAS NOT BEEN MET, GET THE POINTER TO THE     DISTITL    31
C         NEXT TITLE TEXT BEAD. OTHERWISE, RECORD THE NUMBER OF LINES    DISTITL    32
C         WHICH WERE DISPLAYED.                                          DISTITL    33
C                                                                        DISTITL    34
      IF(LINECNT .EQ. 3) GO TO 20                                        DISTITL    35
      CALL DMGET(IWDONE,ITPTR,ITPTR)                                     DISTITL    36
      GO TO 10 .                                                         DISTITL    37
   20 IDTITLE(4) = LINECNT                                               DISTITL    38
      RETURN                                                             DISTITL    39
      END                                                                DISTITL    40
```

```
      SUBROUTINE GAXIS(XINCH,YINCH,XLENIN,THETA,XMIN,DELTA,                GAXIS      2
     X                 ISIGN,IODAD,IDOT,IDOC)                              GAXIS      3
C                                                                          GAXIS      4
C     GAXIS IS A GRAPHICS ROUTINE TO DRAW A LINEAR AXIS.                   GAXIS      5
C                                                                          GAXIS      6
C       XINCH - THE X-COORDINATE IN INCHES OF THE STARTING POINT           GAXIS      7
C               OF THE AXIS.                                               GAXIS      8
C       YINCH - THE Y-COORDINATE IN INCHES OF THE STARTING POINT           GAXIS      9
C               OF THE AXIS.                                               GAXIS     10
C      XLENIN - THE LENGTH OF THE AXIS IN INCHES.                          GAXIS     11
C       THETA - THE ANGLE OF THE AXIS MEASURED IN DEGREES COUNTER -        GAXIS     12
C               CLOCKWISE FROM THE NORMAL POSITIVE X-AXIS. Y CAN           GAXIS     13
C               EQUAL 0., 90., 180., OR 270. DEGREES ONLY.                 GAXIS     14
C        XMIN - THE COORDINATE VALUE AT THE FIRST POINT OF THE AXIS.       GAXIS     15
C       DELTA - COORDINATE UNITS PER INCH ALONG THE AXIS.                  GAXIS     16
C       ISIGN - -1 IF THE COORDINATE UNITS ARE TO BE DISPLAYED ON          GAXIS     17
C               THE CLOCKWISE SIDE OF THE AXIS.                            GAXIS     18
C               +1 IF THE COORDINATE UNITS ARE TO BE DISPLAYED ON          GAXIS     19
C               THE COUNTER - CLOCKWISE SIDE OF THE AXIS.                  GAXIS     20
C       IODAD - THE SYSTEM DEFINED ASSOCIATIVE ADDRESS OF THE              GAXIS     21
C               DISPLAY.                                                   GAXIS     22
C        IDOT - ID DISPLAY TYPE.                                           GAXIS     23
C        IDOC - ID DISPLAY CODE.                                           GAXIS     24
C                                                                          GAXIS     25
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                     CY8777     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                     6JUL77     3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                   /PARMSV/   2
      DATA IFMT1,IFMT2/6H(G9.3),6H(F9.4)/                                  3FE9       2
C                                                                          GAXIS     28
C     USE THETA AND DGUPIN TO CALCULATE THE HORIZONTAL AND VERTICAL        GAXIS     29
C     COMPONENTS OF THE UNIT DISPLACEMENT VECTOR. THE RESULTS ARE IN       GAXIS     30
C     DIGIGRAPHIC UNITS PER INCH.                                          GAXIS     31
C                                                                          GAXIS     32
      THR = .01745329 * THETA                                              GAXIS     33
      STH = SIN(THR)                                                       GAXIS     34
      CTH = COS(THR)                                                       GAXIS     35
      IF(ABS(STH) .LE. .0001) STH = 0.                                     GAXIS     36
      IF(ABS(CTH) .LE. .0001) CTH = 0.                                     GAXIS     37
      IHDEL = IFIX(CTH * DGUPIN + .5)                                      GAXIS     38
      IVDEL = IFIX(STH * DGUPIN + .5)                                      GAXIS     39
C                                                                          GAXIS     40
C     CALCULATE THE COORDINATES OF THE ORIGIN                              GAXIS     41
C     IN DGUS AND DISPLAY A PLUS THERE.                                    GAXIS     42
C                                                                          GAXIS     43
      IXDGU = IFIX(XINCH * DGUPIN + XORDGU + .5)                           GAXIS     44
      IYDGU = IFIX(YINCH * DGUPIN + YORDGU + .5)                           GAXIS     45
      NBYTE = 0                                                            GAXIS     46
      ICODE = 0120                                                         GAXIS     47
      CALL GURSET(IXDGU - 8 ,IYDGU - 12,ICODE,IBUF,NBYTE,MBYTE)            GAXIS     48
      CALL GUAN(1H+,1,IBUF,NBYTE,MBYTE)                                    GAXIS     49
C                                                                          GAXIS     50
C     DRAW THE AXIS AN INCH AT A TIME. FOLLOW EACH ONE INCH LINE           GAXIS     51
C     SEGMENT WITH A PLUS AT EACH NODE. THE CODE TRUNCATES THE             GAXIS     52
C     LENGTH OF THE AXIS TO AN INTEGER.                                    GAXIS     53
C                                                                          GAXIS     54
      NIN = IFIX(XLENIN + .0001)                                           GAXIS     55
      IHI = IXDGU                                                          GAXIS     56
```

```
                      IV1 = IYOGU                                                    GAXIS      57
                      DO 10 I=1, NIN                                                 GAXIS      58
   60                 CALL GURSET(IH1,IV1,ICODE,IBUF,NBYTE,MBYTE)                    GAXIS      59
                      IH2 = IH1 + IHDEL                                              GAXIS      60
                      IV2 = IV1 + IVDEL                                              GAXIS      61
                      CALL GUSEGS(IH1,IV1,IH2,IV2,IBEAM,ISTYLE,IBUF,NBYTE,MBYTE)     GAXIS      62
                      CALL GURSET(IH2 - 8 ,IV2 - 12,ICODE,IBUF,NBYTE,MBYTE)          GAXIS      63
   65                 CALL GUAN(1H+,1,IBUF,NBYTE,MBYTE)                              GAXIS      64
                      IH1 = IH2                                                      GAXIS      65
                      IV1 = IV2                                                      GAXIS      66
                   10 CONTINUE                                                       GAXIS      67
                C                                                                    GAXIS      68
   70           C         CALCULATE THE COORDINATES IN OGUS OF THE POINT ON THE SCREEN GAXIS      69
                C         WHERE THE INITIAL COORDINATE VALUE WILL BE DISPLAYED.      GAXIS      70
                C                                                                    GAXIS      71
                      IDIR = IFIX(THETA/90. + 1.0J0001)                              GAXIS      72
                      GO TO(20,30,20,30),IDIR                                        GAXIS      73
   75                 IXADJ = -100                                                   GAXIS      74
                      IF(ISIGN .EQ. -1) IYADJ = 100 * ISIGN                          GAXIS      75
                      IF(ISIGN .EQ. +1) IYADJ =  50 * ISIGN                          GAXIS      76
                      GO TO 40                                                       GAXIS      77
                   30 IXADJ = - 250                                                  GAXIS      78
   80                 IF(ISIGN .EQ. -1) IXADJ = 50                                   GAXIS      79
                      IYADJ = - 12                                                   GAXIS      80
                   40 IH1 = IXOGU + IXADJ                                            GAXIS      81
                      IV1 = IYOGU + IYADJ                                            GAXIS      82
                   80 NNODES = NIN + 1                                               GAXIS      11
   85                 VAL=XMIN                                                       SCAXES     3
                      IFMT=IFMT2                                                     3FEB       3
                      ABMIN=ABS(XMIN)                                                3FEB       4
                      ADDEL=ABS(DELTA)                                               3FEB       5
                      IF (ABMIN.LT..0001.OR.ABMIN.GT.100.) IFMT=IFMT1                3FEB       6
   90                 IF (ADDEL.GT.100.) IFMT=IFMT1                                  3FEB       7
                      DO 110 I=1,NNODES                                              GAXIS      115
                   90 ENCODE (9,IFMT,VALOC) VAL                                      3FEB       8
                  100 CALL GURSET(IH1,IV1,ICODE,IBUF,NBYTE,MBYTE)                    GAXIS      124
                      CALL GUAN(VALOC,9,IBUF,NBYTE,MBYTE)                            GAXIS      125
   95                 VAL=VAL+DELTA                                                  SCAXES     2
                      IH1 = IH1 + IHDEL                                              GAXIS      128
                      IV1 = IV1 + IVDEL                                              GAXIS      129
                  110 CONTINUE                                                       GAXIS      130
                C                                                                    GAXIS      131
  100           C         DISPLAY THE AXIS.                                          GAXIS      132
                C                                                                    GAXIS      133
                      CALL GIDISP(NCON,IBUF,NBYTE,IODAD,IODT,IODC)                   GAXIS      134
                      ICODE = 1028                                                   GAXIS      135
                      RETURN                                                         GAXIS      136
  105                 END                                                            GAXIS      137
```

220

```
      SUBROUTINE LBLAXES(IPLTPTR)                                        LBLAXES     2
C                                                                        LBLAXES     3
C         THIS ROUTINE LABELS THE X AND Y AXIS OF A PREVIOUSLY DISPLAYED LBLAXES     4
C         AXIS SYSTEM. PRIMARY INPUT TO THE ROUTINE IS THE PLOT BEAD     LBLAXES     5
C         POINTER.                                                       LBLAXES     6
C                                                                        LBLAXES     7
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNVAX,IAXY1, /AXBEAD/  2
     X              IAXY2,IAXY3                                         /AXBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77   2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
      COMMON /GP/ IBUF(128),NCON,NBYTE,IBYTE,IGN,ISIN,                   CYBER77   1
     X     ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77   3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/  2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/  3
     X               IPEMN,IPEMV                                        /PLTBEAD/  4
C                                                                        LBLAXES    12
C         GET THE AXIS NAMES FROM THE AXIS BEAD.                         LBLAXES    13
C                                                                        LBLAXES    14
      CALL DMGET(IPAXIS,IPLTPTR,IAXPTR)                                  LBLAXES    15
      CALL DMGET(IAXXNME,IAXPTR,IXNAME)                                  LBLAXES    16
      CALL DMGET(IAXY1,IAXPTR,IYNAME)                                    LBLAXES    17
C                                                                        LBLAXES    18
C         DISPLAY BOTH AXIS NAMES.                                       LBLAXES    19
C                                                                        LBLAXES    20
      CALL ANNOTA (4.02,-.8,IXNAME,10,IDAXIS(3),IGN,-0,-0,-0)            20SEPT     6
      CALL ANNOTA (-.36,6.37,IYNAME,10,IDAXIS(4),IGN,-0,-0,-0)           20SEPT     7
      RETURN                                                             LBLAXES   23
      END                                                                LBLAXES   24
```

```
      1                SUBROUTINE REPLOF (NCURVE,ICRVPTR,IFITPTR,IFLAG)                  20SEPT      8
                     C                                                                  REPLOF      3
                     C     THIS ROUTINE REPLOTS THE CURVE FITS OF THE                   REPLOF      4
                     C      CURVES WHICH ARE MARKED "INACTIVE".                         REPLOF      5
      5              C                                                                  REPLOF      6
                     C                                                                  REPLOF      7
                      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
                     X              ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
                      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     10               X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),        6JULY77    2
                      X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT    1
                      COMMON /FITBEAD/ IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA             REPLOF     10
                      DIMENSION X(248),Y(248),FP(249)                                   REPLOF     11
                     C                                                                  REPLOF     12
     15              C     QUERY THE CURVE AND CURVE FIT BEADS FOR THE CALL TO DCRVFIT   REPLOF     13
                     C                                                                  REPLOF     14
                      IF (IFLAG.LE.0) GO TO 20                                          20SEPT      9
                      IFITPTR=ICDATAB(NCURVE,4,2)                                       REPLOF     15
                      ICRVPTR=ICDATAB(NCURVE,4,1)                                       REPLOF     16
     20              C                                                                  REPLOF     17
                   20 CALL DMGET (IFTYPE,IFITPTR,ITYPE)                                 13SEPT      7
                      CALL DMGET (ICINDEP,ICRVPTR,IVAR)                                 REPLOF     19
                      CALL DMGET (IFIVVFP,IFITPTR,XINDFP)                               REPLOF     20
                      CALL DMGET (IFIVVLP,IFITPTR,XINDLP)                               REPLOF     21
     25               CALL DMGET (IFIVI,IFITPTR,XINC)                                   REPLOF     22
                     C                                                                  REPLOF     23
                      IF (ITYPE.EQ.6HAITKEN) GO TO 60                                   20SEPT     10
                      CALL DMGET (IFNPARA,IFITPTR,NFTPA)                                REPLOF     24
                      KOMPCOD=7C00000C0068 + SHIFT(NFTPA,18)                            REPLOF     25
     30               CALL DMGET (KOMPCOD,IFITPTR,FP)                                   REPLOF     26
                     C                                                                  REPLOF     27
                     C     IF FIT IS LEAST SQUARES CALL DCRVFIT                         REPLOF     28
                     C     SPLINE NEEDS MORE INFORMATION                                REPLOF     29
                     C                                                                  REPLOF     30
     35               IF (ITYPE.EQ.6HSPLINE) GO TO 50                                   REPLOF     31
                      CALL DCRVFIT (ITYPE,IVAR,XDUM,YDUM,NDUM,FP,NFTPA,XINDFP,XINDLP,   REPLOF     32
                     1     XINC,INITSS,NUMDISP,NCURVE)                                 REPLOF     33
                      GO TO 100                                                         REPLOF     34
                     C                                                                  REPLOF     35
     40              C     SPLINE FIT -- RETRIEVE EXTRA INFORMATION                     REPLOF     36
                     C                                                                  REPLOF     37
                   50 KOMPCOD=KOMPCOD+NFTPA                                             REPLOF     38
                      CALL DMGET (KOMPCOD,IFITPTR,X)                                    REPLOF     39
                      KOMPCOD=KOMPCOD+NFTPA                                             REPLOF     40
     45               CALL DMGET (KOMPCOD,IFITPTR,Y)                                    REPLOF     41
                     C                                                                  REPLOF     42
                      CALL DCRVFIT (ITYPE,IVAR,X,Y,NFTPA,FP,NFTPA,XINDFP,XINDLP,        REPLOF     43
                     1     XINC,INITSS,NUMDISP,NCURVE)                                 REPLOF     44
                      GO TO 100                                                         20SEPT     11
     50              C                                                                  20SEPT     12
                     C          AITKEN FIT -- RETRIEVE X AND Y ARRAYS                   20SEPT     13
                     C                                                                  20SEPT     14
                   60 CALL DMGET (IFNPARA,IFITPTR,NPOINTS)                              20SEPT     15
                      CALL DMGET (IFNPARA+1,IFITPTR,NFTPA)                              5OCT       5
     55               KOMPCOD=7C0000C0078+SHIFT(NPOINTS,18)                             20SEPT     17
                      CALL DMGET (KOMPCOD,IFITPTR,X)                                    20SEPT     18
                      CALL DMGET (KOMPCOD+NPOINTS,IFITPTR,Y)                            20SEPT     19
```

```
      C                                                                  20SEPT   20
            CALL OCRVFIT (ITYPE,IVAR,X,Y,NPOINTS,FP,NFTPA,XINOFP,XINOLP,  20SEPT   21
     60   1     XINC,INITSS,NUMDISP,NCURVE)                              20SEPT   22
      C                                                                  REPLCF   45
      C     UPDATE CURVE DISPLAY TABLE                                   REPLCF   46
      C                                                                  REPLCF   47
        100 CALL UCDATAB (NCURVE,2,6HACTIVE,INITSS,NUMDISP,IFITPTR)      REPLCF   48
     65     RETURN                                                       REPLCF   49
            END                                                          REPLCF   50
```

```
      SUBROUTINE REPLOT                                                 REPLOT     2
C                                                                       REPLOT     3
C       THIS ROUTINE SEARCHES THROUGH THE CURVE DISPLAY ACCESS TABLE    REPLOT     4
C       AND REDISPLAYS ALL CURVES LISTED "INACTIVE" IN THE TABLE.       REPLOT     5
C                                                                       REPLOT     6
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/  2
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IDCO(23),IDMENU(20),         6JULY77    2
     X               IOINSTR(19),ICDATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT     1
C                                                                       REPLOT     9
C       LOOP THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE. IF A   REPLOT    10
C       CURVE IS ACTIVE, INCREASE THE IODAD INDEX(NN) BY THE NUMBER OF  REPLOT    11
C       ELEMENTS OF IODAD REQUIRED TO DISPLAY IT AND THEN GO TO THE     REPLOT    12
C       NEXT ONE. IF A CURVE IS INACTIVE, THEN DISPLAY IT AND ITS       REPLOT    13
C       DESCRIPTION AND UPDATE THE CURVE DISPLAY ACCESS TABLE.          REPLOT    14
C                                                                       REPLOT    15
      CALL ORIGIN(-5.,-3.)                                              5MAY76     7
      NN = 0                                                            REPLOT    16
      DO 100 I=1,NCIP                                                   3JUNE76    1
      IF(ICDATA-(I,1,1) .EQ. 8HINACTIVE) GO TO 10                       14APR76    5
      NN = NN + ICDATAB(I,3)                                            REPLOT    19
      GO TO 30                                                          REPLOT    20
   10 CALL DISCRVE (ICDATAB(I,4,1),ISYMBOL,INITSS,NUMDISP,I)            ADPTPRE    7
      IF(NUMDISP .GT. 0) GO TO 20                                       REPLOT    22
C                                                                       REPLOT    23
C       THIS CODE IS HERE TO COVER THE POSSIBILITY THAT EVERY POINT ON  REPLOT    24
C       A CURVE HAS BEEN DELETED. THE CURVE IS ACTIVE BUT EMPTY.        REPLOT    25
C                                                                       REPLOT    26
      NN = NN + 1                                                       REPLOT    27
      IODAD(NN) = 0                                                     REPLOT    28
      CALL DCRVDES (ICDATAB(I,4,1),ISYMBOL,I)                           25MAY76    4
      CALL UCDATAB(I,1,6HACTIVE,INITSS,1,ICDATAB(I,4,1))                14APR76    6
      GO TO 30                                                          REPLOT    31
   20 CALL DCRVDES (ICDATAB(I,4,1),ISYMBOL,I)                           25MAY76    5
      CALL UCDATAB(I,1,6HACTIVE,INITSS,NUMDISP,ICDATAB(I,4,1))          14APR76    7
   30 IF (ICDATAB(I,4,2).LE.0) GO TO 100                                3JUNE76    2
      IF (ICDATAB(I,1,2).EQ.6HACTIVE) GO TO 100                         3JUNE76    3
      CALL REPLOF (I,I1,I2,1)                                           20SEPT    23
  100 CONTINUE                                                          3JUNE76    5
      RETURN                                                            REPLOT    35
      END                                                               REPLOT    36
```

```
      SUBROUTINE UCDATAB(NROW,NPAGE,ISTATUS,INITSS,NUMDISP,IBEAD)          UCDATAB    2
C                                                                          UCDATAB    3
C     THIS ROUTINE UPDATES THE CURVE DISPLAY ACCESS TABLE.                 UCDATAB    4
C                                                                          UCDATAB    5
C                *** THE CURVE DISPLAY ACCESS TABLE ***                    UCDATAB    6
C     THE CURVE DISPLAY ACCESS TABLE IS USED TO STORE ACCESS               UCDATAB    7
C     INFORMATION ABOUT THE CURVES AND/OR CURVE FITS WHICH BELONG TO       UCDATAB    8
C     THE PLOT ""WHICH IS DISPLAYED ON THE SCREEN"". THE INFORMATION       UCDATAB    9
C          ************************************                            UCDATAB   10
C     IN THE TABLE CAN BE USED TO:                                         UCDATAB   11
C          1 - DETERMINE IF A CURVE OR A CURVE FIT FOR A CURVE IN          UCDATAB   12
C              THE PLOT IS DISPLAYED ON THE SCREEN.                        UCDATAB   13
C          2 - ERASE A CURVE OR CURVE FIT IN THE PLOT, OR                  UCDATAB   14
C          3 - RETURN TO THE CURVE BEAD OR CURVE FIT BEAD(IF IT            UCDATAB   15
C              EXISTS) OF A CURVE IN THE PLOT.                             UCDATAB   16
C     THE I-TH ROW OF ICDATAB IS INFORMATION ABOUT THE I-TH CURVE          UCDATAB   17
C     (CHRONOLOGICALLY) TO BE DISPLAYED ON THE SCREEN.                     UCDATAB   18
C                    ** PAGE ONE---CURVE INFORMATION **                    UCDATAB   19
C     ICDATAB(I,1,1) IS THE STATUS OF THE I-TH CURVE. IF                   UCDATAB   20
C                    ICDATAB(I,1,1) = 6HACTIVE, THEN THE CURVE IS          UCDATAB   21
C                    CURRENTLY DISPLAYED ON THE SCREEN. IF                 UCDATAB   22
C                    ICDATAB(I,1,1) = 8HINACTIVE, THEN THE CURVE HAS       UCDATAB   23
C                    BEEN DISPLAYED ON THE SCREEN AND ERASED.              UCDATAB   24
C     ICDATAB(I,2,1) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB   25
C                    CURVE. INITSS IS EXPLAINED IN SUBROUTINE              UCDATAB   26
C                    DISCRVE.                                              UCDATAB   27
C     ICDATAB(I,3,1) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB   28
C                    EXHIBIT THE I-TH CURVE.                               UCDATAB   29
C     ICDATAB(I,4,1) IS THE POINTER TO THE CURVE BEAD OF THE I-TH          UCDATAB   30
C                    CURVE.                                                UCDATAB   31
C                  ** PAGE TWO---CURVE FIT INFORMATION **                  UCDATAB   32
C     ICDATAB(I,1,2) IS THE STATUS OF THE I-TH CURVE FIT. IF               UCDATAB   33
C                    ICDATAB(I,1,2) = 4HNONE, THEN THE CURVE DOES          UCDATAB   34
C                    NOT HAVE A CURVE FIT. IF ICDATAB(I,1,2) =             UCDATAB   35
C                    6HACTIVE, THEN THE CURVE HAS A CURVE FIT AND IT       UCDATAB   36
C                    IS DISPLAYED ON THE SCREEN. IF ICDATAB(I,1,2) =       UCDATAB   37
C                    8HINACTIVE, THEN THE CURVE HAS A CURVE FIT BUT        UCDATAB   38
C                    IT IS NOT DISPLAYED ON THE SCREEN.                    UCDATAB   39
C     ICDATAB(I,2,2) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB   40
C                    CURVE FIT. INITSS IS EXPLAINED IN SUBROUTINE          UCDATAB   41
C                    DCRVFIT.                                              UCDATAB   42
C     ICDATAB(I,3,2) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB   43
C                    EXHIBIT THE I-TH CURVE FIT.                           UCDATAB   44
C     ICDATAB(I,4,2) IS THE POINTER TO THE FIT BEAD OF THE I-TH            UCDATAB   45
C                    CURVE.                                                UCDATAB   46
      COMMON/DISPLAY/IDSTOP,IDDAD(91),NN,NNEND(10),IDSPEC(19),IDCF(41),    /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCC(23),IDMENU(20),             6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM    1SEPT      1
      ICDATAB(NROW,1,NPAGE) = ISTATUS                                      UCDATAB   48
      ICDATAB(NROW,2,NPAGE) = INITSS                                       UCDATAB   49
      ICDATAB(NROW,3,NPAGE) = NUMDISP                                      UCDATAB   50
      ICDATAB(NROW,4,NPAGE) = IBEAD                                        UCDATAB   51
      RETURN                                                               UCDATAB   52
      END                                                                  UCDATAB   53
```

```
      FUNCTION P(Z,C,N)                                                 P          2
C                                                                       P          3
C         THIS FOUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES P          4
C         POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".          P          5
C                                                                       P          6
C         Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                P          7
C         C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.   P          8
C         N - I - THE NUMBER OF COEFFICIENTS.                           P          9
C                                                                       P         10
      DIMENSION C(N)                                                    P         11
      NDEG = N - 1                                                      P         12
      P=C(1)                                                            18JUNE76  21
      DO 10 I=2,N                                                       18JUNE76  22
      P=P+C(I)*Z**(I-1)                                                 18JUNE76  23
   10 CONTINUE                                                          P         16
      RETURN                                                            P         17
      END                                                               P         18
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C - - A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE       SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 8 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 8                                          SPLIN2    27
       KK = K                                                            SPLIN2    28
       GO TO 120                                                         SPLIN2    29
    8 CONTINUE                                                           SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN                                                             SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1       1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3    SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
      RETURN                                                             SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
      SUBROUTINE OCRVFIT(ITYPE,IV,X,Y,NPOINTS,C,NFP,TMIN,TMAX,TINCR,     OCRVFIT     2
     X          INITSS,NUMDISP,ICURVE)                                   28MAY76     1
C                                                                       OCRVFIT     4
C     THIS ROUTINE DISPLAYS THE CURVE FIT DEFINED BY PARAMETERS ONE     OCRVFIT     5
C     THROUGH SEVEN OVER THE INTERVAL DEFINED BY PARAMETERS EIGHT       OCRVFIT     6
C     AND NINE.                                                         OCRVFIT     7
C               SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND             13SEPT     18
C               6HAITKEN FOR A STRAIGHT FIT.                            13SEPT     19
C               SQUARES FIT AND 6HSPLINE FOR A SPLINE FIT.              OCRVFIT     9
C          IV - I - THE INDEPENDENT VARIABLE; EITHER 1HX OR 1HY.        OCRVFIT    10
C           X - I - THE ARRAY OF X-COORDINATES OF THE POINTS FOR        OCRVFIT    11
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      OCRVFIT    12
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     OCRVFIT    13
C           Y - I - THE ARRAY OF Y-COORDINATES OF THE POINTS FOR        OCRVFIT    14
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      OCRVFIT    15
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     OCRVFIT    16
C     NPOINTS - I - THE NUMBER OF POINTS WHOSE COORDINATES              OCRVFIT    17
C               ARE IN ARRAYS "X" AND "Y".                              OCRVFIT    18
C           C - I - THE ARRAY OF CURVE FIT PARAMETERS; POLYNOMIAL       OCRVFIT    19
C               COEFFICIENTS FOR A LEAST SQUARES FIT, DERIVATIVES       OCRVFIT    20
C               AT EACH POINT FOR A SPLINE FIT.                         OCRVFIT    21
C         NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF         OCRVFIT    22
C               POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT,        OCRVFIT    23
C               THE NUMBER OF POINTS FOR A SPLINE FIT.                  OCRVFIT    24
C        TMIN - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      OCRVFIT    25
C               THE DISPLAY OF THE CURVE FIT IS TO START.               OCRVFIT    26
C        TMAX - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      OCRVFIT    27
C               THE DISPLAY OF THE CURVE FIT IS TO STOP. (TMIN,TMAX)    OCRVFIT    28
C               IS THE INTERVAL OVER WHICH THE CURVE FIT IS TO BE       OCRVFIT    29
C               DISPLAYED.                                              OCRVFIT    30
C        TINCR - I - THE INCREMENT OF THE INDEPENDENT VARIABLE PER      OCRVFIT    31
C               TWENTIETH OF AN INCH. THE CURVE FIT WILL BE             OCRVFIT    32
C               APPROXIMATED BY TWENTY STRAIGHT LINES PER INCH.         OCRVFIT    33
C       INITSS - O - THE DISPLAY ADDRESSES OF THE CURVE FIT ARE         OCRVFIT    34
C               STORED IN ARRAY "IDCF". IDCF(INITSS) CONTAINS THE       OCRVFIT    35
C               DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED TO        OCRVFIT    36
C               EXHIBIT THE CURVE FIT.                                  OCRVFIT    37
C     NUMDISP - O - NUMDISP CONTAINS THE NUMBER OF DISPLAYS             OCRVFIT    38
C               REQUIRED TO EXHIBIT THE CURVE FIT.                      OCRVFIT    39
C                                                                       OCRVFIT    40
C     A 64 WORD DESCRIPTION BUFFER WILL HOLD THE DESCRIPTION BYTES      OCRVFIT    41
C     FOR 101 LINE SEGMENTS.                                            OCRVFIT    42
C          NUMBER     TYPE      BYTES                                   OCRVFIT    43
C            OF        OF        PER        TOTAL                       OCRVFIT    44
C          CALLS      CALL      CALL       BYTES                       OCRVFIT    45
C          ******     ******    *******    *******                     OCRVFIT    46
C            1        GURSET       3          3                        OCRVFIT    47
C            1        GUSEGI       4          4                        OCRVFIT    48
C          101        GUSEG        3        303                        OCRVFIT    49
C            1        GIDISP      10         10                        OCRVFIT    50
C                                           320---TOTAL                 OCRVFIT    51
C     SINCE ANY CURVE FIT CAN BE DISPLAYED WITH AT MOST 180 LINE        OCRVFIT    52
C     SEGMENTS(NINE INCHES), EACH CURVE FIT CAN BE DISPLAYED WITH       OCRVFIT    53
C     AT MOST TWO BUFFERS.                                              OCRVFIT    54
C                                                                       OCRVFIT    55
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/   2
      COMMON/DISPLAY/IOSTOP,IDDA(40),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/   2
```

```
                X                IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),                6JULY77     2
                X                IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM       1SEPT       1
   60           COMMON /GF/ IBUF(128),NCON,N3YTE,MBYTE,IGN,ISIN,                         CYBTTT      1
                X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                       6JULY77     3
                COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                       /PARMSV/    2
                COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,         /CRVBEAD/   2
                X                ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                      /CRVBEAD/   3
   65           DIMENSION TEMP(14)                                                       13SEPT      20
                DIMENSION X(NPOINTS),Y(NPOINTS),C(NFP)                                   DCRVFIT     60
          C                                                                              DCRVFIT     61
          C         RECORD THE INITIAL VALUE OF THE "IDCF" SUBSCRIPT.                    DCRVFIT     62
          C                                                                              DCRVFIT     63
   70           N=(ICURDF-1)*2                                                           28MAY76     2
                INITSF = N + 1                                                           DCRVFIT     65
          C                                                                              DCRVFIT     66
                IF(IV .EQ. 1HY) GO TO 50                                                 DCRVFIT     67
                IF(IV .EQ. 1HX) CONTINUE                                                 DCRVFIT     68
   75           C                                                                       DCRVFIT     69
          C         INITIALIZE THE DESCRIPTION BUFFER TO THE FIRST POINT ON THE          DCRVFIT     70
          C         CURVE FIT. THE Y-COORDINATE OF THE POINT DEPENDS UPON THE            DCRVFIT     71
          C         TYPE OF CURVE FIT.                                                   DCRVFIT     72
          C                                                                              DCRVFIT     73
   80           XX = XMIN                                                               DCRVFIT     74
                XXINCR = XINCR                                                           DCRVFIT     75
                XXMAX = XMAX                                                             DCRVFIT     76
229            2 IF (ITYPE.EQ.1HLEASTSQUAR) YY=P(XX,C,NFP)                              29APR       2
                IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)                      DCRVFIT     77
   85           IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,                 20SEPT      78
                1     YY,TEMP,IER)                                                       13SEPT      22
                IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX)             29APR       3
                1     GO TO 5                                                            27MAY       3
                XX=XX+XXINCR                                                             29APR       5
   90           IF (XX.GT.XXMAX) GO TO 32                                                29APR       6
                GO TO 2                                                                  29APR       7
                5 IF (IER.EQ.2) GO TO 100                                                29APR       8
                IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)                    DCRVFIT     79
                IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)                    DCRVFIT     80
   95           NBYTE = 0                                                               DCRVFIT     81
               10 CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                             DCRVFIT     82
                CALL GUSEGI(IX,IY,TSTYLE,IBUF,NBYTE,MBYTE)                               DCRVFIT     83
          C                                                                              DCRVFIT     84
          C         INCREMENT THROUGH THE INTERVAL AND PACK THE DESCRIPTION BYTES        DCRVFIT     85
  100     C         OF APPROXIMATING LINE SEGMENTS INTO THE DESCRIPTION BUFFER.          DCRVFIT     86
          C                                                                              DCRVFIT     87
               20 XX = XX + XXINCR                                                       DCRVFIT     88
                IF(ITYPE .EQ. 1HLEASTSQUAR) YY = P(XX,C,NFP)                             DCRVFIT     89
                IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)                      DCRVFIT     90
  105           IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,                 20SEPT      77
                1     YY,TEMP,IER)                                                       13SEPT      24
                IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 30                                   14MARCH     17
                IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 30                                   14MARCH     18
                IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)                    DCRVFIT     91
  110           IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)                    DCRVFIT     92
                CALL GUSEG(IX,IY,IBEAM)                                                  DCRVFIT     93
          C                                                                              DCRVFIT     94
          C         CHECK TO SEE IF THE DESCRIPTION BUFFER IS FULL.                      DCRVFIT     95
          C         IF SO, FLUSH IT. IF NOT, CONTINUE TO FILL IT.                        DCRVFIT     96
```

```
115          C                                                                  14APR76   71
                  IF(NBYTE .LT. 312) GO TO 30                                   OCRVFIT   97
                  N = N + 1                                                     OCRVFIT   98
                  CALL GIDISP(NCON,IBUF,NBYTE,IOCF(N),IGN)                      OCRVFIT   99
                  IF(XX .LT. XXMAX) GO TO 10                                    OCRVFIT  100
120          C                                                                  OCRVFIT  101
             C       THE DESCRIPTION BUFFER IS NOT FULL. CHECK TO SEE IF THE CURVE OCRVFIT  102
             C        FIT IS COMPLETE. IF NOT, CONTINUE TO GENERATE THE DESCRIPTION OCRVFIT  103
             C        BYTES FOR THE CURVE FIT. IF SO, DISPLAY THE CURVE FIT.    OCRVFIT  104
             C                                                                  OCRVFIT  105
125             30 IF(XX .LT. XXMAX) GO TO 20                                   OCRVFIT  106
                32 IF (NBYTE.EQ.0) GO TO 40                                     29APR     9
                   N = N + 1                                                    OCRVFIT  108
                   CALL GIDISP(NCON,IBUF,NBYTE,IOCF(N),IGN)                     OCRVFIT  109
                40 NUMDISP = N - INITSS + 1                                     OCRVFIT  110
130                IOCF(41)=N                                                   25MAY76   15
                   RETURN                                                       OCRVFIT  111
             C                                                                  OCRVFIT  112
             C       "Y" IS THE INDEPENDENT VARIABLE. COMMENTS                  OCRVFIT  113
             C       ANALOGOUS TO THOSE ABOVE APPLY.                            OCRVFIT  114
135          C                                                                  OCRVFIT  115
                50 YY = YMIN                                                    OCRVFIT  116
                   YYINCR = YINCR                                               OCRVFIT  117
                   YYMAX = YMAX                                                 OCRVFIT  118
                52 IF (ITYPE.EQ.10HLEASTSQUAR) XX=P(YY,C,NFP)                   29APR     10
230 140            IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)          OCRVFIT  120
                   IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,     20SEPT    78
                  1    XX,TEMP,IER)                                            13SEPT    26
                   IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX) 29APR     11
                  1       GO TO 55                                              29APR     12
145                YY=YY+YYINCR                                                 29APR     13
                   IF (YY.GT.YYMAX) GO TO 82                                    29APR     14
                   GO TO 52                                                     29APR     15
                55 IF (IER.EQ.2) GO TO 100                                      29APR     16
                   IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)        OCRVFIT  121
150                IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)        OCRVFIT  122
                   NBYTE = 0                                                    OCRVFIT  123
                60 CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                   OCRVFIT  124
                   CALL GUSEGI(IX,IY,ISTYLE,IBUF,NBYTE,MBYTE)                  OCRVFIT  125
                70 YY = YY + YYINCR                                             OCRVFIT  126
155                IF(ITYPE .EQ. 10HLEASTSQUAR) XX = P(YY,C,NFP)                OCRVFIT  127
                   IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)          OCRVFIT  128
                   IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,     20SEPT    79
                  1    XX,TEMP,IER)                                            13SEPT    28
                   IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 80                       14MARCH   21
160                IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 80                       14MARCH   22
                   IF (IER.EQ.2) GO TO 100                                      3MARCH    19
                   IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)        OCRVFIT  129
                   IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)        OCRVFIT  130
                   CALL GUSEG(IX,IY,IBEAM)                                      OCRVFIT  131
165                IF(NBYTE .LT. 312) GO TO 90                                  OCRVFIT  132
                   N = N + 1                                                    OCRVFIT  133
                   CALL GIDISP(NCON,IBUF,NBYTE,IOCF(N),IGN)                     OCRVFIT  134
                   IF(YY .LT. YYMAX) GO TO 60                                   OCRVFIT  135
                80 IF(YY .LT. YYMAX) GO TO 70                                   OCRVFIT  136
170                82 IF (NBYTE.EQ.0) GO TO 90                                  29APR     17
                   N = N + 1                                                    OCRVFIT  138
```

```
                  CALL GIDISP(NCON,I8UF,N8YTE,IDCF(N),IGN)                          DCRVFIT   139
               90 NUMDISP = N - INITSS + 1                                          DCRVFIT   140
                  IDCF(41)=N                                                        25MAY76    16
175               RETURN                                                           DCRVFIT   141
C                                                                                  3MARCH     20
C        ERROR IN AITKEN CALL - RETURN WITH NO FIT                                 3MARCH     21
C                                                                                  3MARCH     22
             100 NCF=(ICURVE-1)*2+1                                                3MARCH     23
180               CALL GIESAS (IDCF(NCR),IDCF(NCR+1))                              3MARCH     24
                  IDCF(NCR)=IDCF(NCR+1)=0                                          3MARCH     25
                  IF (ICDATA8(ICURVE,4,2).NE.0) CALL DMRL8D (ICDATA8(ICURVE,4,2))  3MARCH     26
                  ICDATA8(ICURVE,4,2)=0                                            3MARCH     27
                  CALL DMSET (ICFIT,ICDATA8(I,4,1),0)                              3MARCH     28
185               CALL ERROKM (1.,3.5,41HINDEPENDENT VARIABLE ERROR WITH CURVE FIT, 3MARCH     29
                 1    41,7HDCRVFIT,6HINFORM)                                       3MARCH     30
                  RETURN                                                           3MARCH     31
                  END                                                              DCRVFIT   142
```

231

```
      1                OVERLAY(4,0)                                                          QTRBACK     2
                       PROGRAM QTRBACK                                                       QTRBACK     3
                 C                                                                           QTRBACK     4
                 C         THIS ROUTINE DISPLAYS THE CROSS PLOT OR TRIM MENU (WHICHEVER      QTRBACK     5
      5          C         IS APPROPRIATE) AND CALLS THE ROUTINE INDICATED BY THE USERS      QTRBACK     6
                 C         PICK. PRIMARY INPUTS TO THE ROUTINE ARE "IOPTION" WHICH           QTRBACK     7
                 C         IS AVAILABLE IN COMMON/OPTION/ AND "IPLTPTR" WHICH IS             QTRBACK     8
                 C         AVAILABLE IN COMMON/BEADS/.                                       QTRBACK     9
                 C                                                                           QTRBACK    10
     10          COMMON/BEADS/IFGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR          /BEADS/     2
                 COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),          /DISPLAY/   2
                X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),                  6JULY77     2
                X               IDINSTR(19),IDDATA3(20,4,2),NCIP,IDBLOCK(20),NCIPN         1SEPT       1
                 COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                          CYBER77     1
     15         X     ISTR,ABUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                        6JULY77     3
                 COMMON/OPTION/IOPTION,ISHINE                                              LASTBUX     1
                 COMMON /NXS/ NEXTSET,IERR                                                 NEWFACE    25
                 LOGICAL NEXTSET                                                           INHALE     13
                 NEXTSET=.FALSE.                                                           INHALE     14
     20          IF (IDMENU(1).GT.0) GO TO 10                                              25JULY77   10
                 CALL RETSYC (4HMENU)                                                      FIXINH      9
                 C                                                                         QTRBACK    16
                 C         WAIT FOR A BUTTON PICK FROM ONE OF THE FOLLOWING MENUS.         QTRBACK    17
                 C              CROSS PLOT MENU          X          TRIM MENU             QTRBACK    18
     25          C                                                                         QTRBACK    19
                 C                1 - EDIT               X    1 - EDIT                    QTRBACK    20
                 C                2 - CURVE FIT          X    2 - CURVE FIT               QTRBACK
                 C                3 - HARDCOPY           X    3 - HARDCOPY                QTRBACK
                 C                4 - NEXT PLOT          X    4 - NEXT PLOT               QTRBACK
                 C                5 - ANALYZE            X    7 - NEXT SET                QTRBACK    23
     30          C                6 - RETURN TO PLOT GROUP X    8 - COMPUTE POLAR           QTRBACK    24
                 C                    SPECIFICATIONS     X    9 - RETURN TO TABLE         QTRBACK    25
                 C               10 - CHANGE SCALE       X    6 - RETURN TO PLOT GROUP    26AUG       1
                 C              254 - STOP               X         SPECIFICATIONS         26AUG       2
                 C                                       X   10 - CHANGE SCALE            26AUG       3
     35          C                                       X  254 - STOP                    14APR76    30
                 C                                                                         QTRBACK    29
                    10 CALL GIBUT(0,NCON,IDOT,IDDC)                                        QTRBACK    30
                       CALL EMBER                                                          LASTBUG     3
                       IF(IDDC .EQ. 1) GO TO 20                                            QTRBACK    31
     40                IF(IDDC .EQ. 2) GO TO 30                                            QTRBACK    32
                       IF(IDDC .EQ. 3) GO TO 40                                            QTRBACK    33
                       IF(IDDC .EQ. 4) GO TO 50                                            QTRBACK    34
                       IF(IDDC .EQ. 5) GO TO 60                                            QTRBACK    35
                       IF(IDDC .EQ. 6) GO TO 70                                            QTRBACK    36
     45                IF(IDDC .EQ. 7) GO TO 80                                            QTRBACK    37
                       IF(IDDC .EQ. 8) GO TO 90                                            QTRBACK    38
                       IF(IDDC .EQ. 9) GO TO 100                                           QTRBACK    39
                       IF(IDDC .EQ. 10) GO TO 110                                          26AUG       4
                       IF (IDDC.EQ.11) GO TO 120                                           AXNME       1
     50                IF (IDDC.EQ.12) GO TO 130                                           13APR       1
                       IF (IDDC.EQ.13) GO TO 140                                           10MAY77     3
                       IF (IDDC.EQ.14) GO TO 150                                           14JUNE     15
                       IF (IDDC.EQ.15) GO TO 160                                           14JUNE     16
                       IF(IDDC .EQ. 254) CALL STOP                                         14APR76    31
     55                GO TO 10                                                            QTRBACK    41
                 C                                                                         QTRBACK    42
                 C         EXIT WAS SELECTED.                                              QTRBACK    43
```

```
                C                                                                 QTRBACK    44
                   20 CALL TERASE (IDMENU,IDMENU(20))                             6JULY77    12
       60             CALL AETSKC(4HEDIT)                                        QTRBACK    46
                C                                                                 QTRBACK    47
                C        CURVE FIT WAS SELECTED.                                  QTRBACK    48
                C                                                                 QTRBACK    49
                   30 CALL TERASE (IDMENU,IDMENU(20))                             6JULY77    13
       65             CALL AETSKC(7HCRVEFIT)                                     QTRBACK    51
                C                                                                 QTRBACK    52
                C        HARDCOPY WAS SELECTED.                                   QTRBACK    53
                C                                                                 QTRBACK    54
                   40 CALL TWINKLE (1)                                            FIXINH     10
       70          CALL HARDCPY                                                   FIXINH     11
                   CALL AETSKC (5HPRNTX)                                          INHALE     15
                   GO TO 10                                                       QTRBACK    56
                C                                                                 QTRBACK    57
                C        NEXT PLOT WAS SELECTED.                                  QTRBACK    58
       75       C                                                                 QTRBACK    59
                   50 CALL AETSKC (7HNXTPLOT)                                     INHALE     16
                   IF (IERR.NE.1) CALL AETSKC(7HDISPLTX)                          NEWFACE    26
                   GO TO 10                                                       QTRBACK    61
                C                                                                 QTRBACK    62
       80       C        ANALYZE WAS SELECTED.                                    QTRBACK    63
                C                                                                 QTRBACK    64
233                60 CALL TERASE (IDMENU,IDMENU(20))                             6JULY77    14
                   CALL AETSKC (6HCROSSX)                                         FIXINH     12
                   GO TO 10                                                       LASTBUX     2
       85       C                                                                 QTRBACK    67
                C        "RETURN TO PLOT GROUP SPECIFICATIONS" WAS SELECTED.      QTRBACK    68
                C                                                                 QTRBACK    69
                   70 CALL WINDEX                                                 QTRBACK    70
                   CALL AETSKC(7HPGSPECS)                                         QTRBACK    71
       90       C                                                                 QTRBACK    72
                C        NEXT SET WAS SELECTED.                                   QTRBACK    73
                C                                                                 QTRBACK    74
                   80 NEXTSET=.TRUE.                                              INHALE     18
                   CALL AETSKC (7HNXTPLOT)                                        FIXINH     13
       95          CALL AETSKC (7HDISPLTX)                                        INHALE     20
                   GO TO 10                                                       QTRBACK    76
                C                                                                 QTRBACK    77
                C        COMPUTE POLAR WAS SELECTED.                              QTRBACK    78
                C                                                                 QTRBACK    79
      100          90 CALL WINDEX                                                 QTRBACK    80
                   CALL AETSKC (5HDTRIM)                                          INITR      14
                C                                                                 QTRBACK    82
                C        "RETURN TO TABLE" WAS SELECTED.                          QTRBACK    83
                C                                                                 QTRBACK    84
      105        100 CALL WINDEX                                                 QTRBACK    85
                   CALL AETSKC(5HTABLE)                                           QTRBACK    86
                C                                                                 27AUG       1
                C        "CHANGE SCALE" WAS SELECTED.                             27AUG       2
                C                                                                 27AUG       3
      110        110 CALL RESCALE (IPLTPTR)                                      27AUG       4
                   CALL WINDEX                                                    26AUG       6
                   CALL AETSKC(7HFRSTPL2)                                         10MAY77     8
                 120 CALL CHAYNM                                                  AXNME       2
                   GO TO 10                                                       AXNME       3
```

```
115          C                                                          13APR      2
             C        "CROSS PLOT TWO" WAS SELECTED                     13APR      3
             C                                                          13APR      4
               13C CALL TERASE (IOMENU,IOMENU(20))                      6JULY77   15
                   CALL AETSKC (7HCROSS2X)                              FIXINH    14
120                GO TO 10                                             LASTBUX    3
             C                                                          10MAY77   13
             C          "PREFIT REMAINING CURVES" WAS SELECTED.         10MAY77   14
             C                                                          10MAY77   15
               14C CALL WINDEX                                          10MAY77   16
125                CALL AETSKC(7HFRSTPLT)                               10MAY77   17
             C                                                          14JUNE    17
             C        "CHANGE TITLE" WAS SELECTED                       14JUNE    18
             C                                                          14JUNE    19
               15C CALL CHTINM (IPLTPTR)                                14JUNE    20
130                GO TO 10                                             14JUNE    21
             C                                                          14JUNE    22
             C        "PRINT PLOT INFORMATION" WAS SELECTED             14JUNE    23
             C                                                          14JUNE    24
               16C CALL TWINKLE (2)                                     FIXINH    15
135                CALL AETSKC(5HPRNTX)                                 FIXINH    16
                   GO TO 10                                             14JUNE    26
                   END                                                  QTRBACK   87
```

```
      FUNCTION ALPHIN(TEXT,NC,TITLE,NCMX,IX,IY,NCH)                    ALPHIN      2
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                 CY8777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                6JULY77     3
      DIMENSION TITLE(1),TEXT(1)                                       ALPHIN      4
C     WRITES A MESSAGE STARTING AT IX,IY FOLLOWED BY A 10 CHARACTER    ALPHIN      5
C     LIGHT REGISTER                                                   ALPHIN      6
      CALL GIEOM (NCON,IEOM,IBUT,200)                                  CY8777      7
      CALL GURSET(IX,IY,0028,IBUF,NBYTE,MBYTE)                         ALPHIN      8
      CALL GUAP(TEXT,NC,IBUF,NBYTE,MBYTE)                              ALPHIN      9
      CALL GIDISP(NCON,IBUF,NBYTE,IDDAD,1,101)                         ALPHIN     10
      NBYTE=0                                                          ALPHIN     11
      IH=IX+NC*24                                                      ALPHIN     12
   91 CALL GIANE(NCON,NCMX,IH,IY)                                      ALPHIN     13
   10 CALL GIBUT(0,NCON,IDDT,IDDC)                                     ALPHIN     14
      IF(IDDC.NE.200)GO TO 10                                          ALPHIN     15
      NCH=NCMX                                                         27MAY       2
      CALL GIANE(NCON,NCH,TITLE)                                       ALPHIN     17
      IF(NCH.LT.1)GO TO 91                                             ALPHIN     20
      CALL GIERAS(IDDAD)                                               ALPHIN     21
      ALPHIN=TITLE(1)                                                  ALPHIN     22
      END                                                              ALPHIN     23
```

```
      SUBROUTINE CHAXNM                                                 CHAXNM    2
C                                                                       CHAXNM    3
C        THIS ROUTINE ENABLES THE OPERATOR TO CHANGE                    CHAXNM    4
C        THE AXIS NAMES AT ANY TIME                                     CHAXNM    5
C                                                                       CHAXNM    6
      DIMENSION NAMES(2),XNMS(2),ISCR(3)                                CHAXNM    7
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1, /AXBEAD/  2
     X              IAXY2,IAXY3                                         /AXBEAD/  3
      COMMON/DISPLAY/IOSTOP,IODA0(90),NN,NNENO(10),IOSPEC(19),IOCF(41), /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),         6JULY77   2
     X               IOINSTR(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT    1
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/  2
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,      /PLTBEAD/  2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/  3
     X               IPFMN,IPFMV                                        /PLTBEAD/  4
C                                                                       CHAXNM   12
C        GET NEW NAMES FROM DISPLAY                                     CHAXNM   13
C                                                                       CHAXNM   14
      NAMES(1)=10HX AXIS                                                CHAXNM   15
      NAMES(2)=10HY AXIS                                                CHAXNM   16
      CALL DMGET (IPAXIS,IPLTPTR,IAXPTR)                                CHAXNM   17
      CALL DMGET (IAXXNME,IAXPTR,XNMS(1))                               CHAXNM   18
      CALL DMGET(IAXY1,IAXPTR,XNMS(2))                                  CHAXNM   19
C                                                                       CHAXNM   20
      CALL VARIAN (NAMES,XNMS,2,-1000,500,ISCR)                         CHAXNM   21
C                                                                       CHAXNM   22
C        RESET AXIS NAMES AND REPLOT                                    CHAXNM   23
C                                                                       CHAXNM   24
      CALL DMSET (IAXXNME,IAXPTR,XNMS(1))                               CHAXNM   25
      CALL DMSET (IAXY1,IAXPTR,XNMS(2))                                 CHAXNM   26
C                                                                       CHAXNM   27
      CALL GIERAS (IOAXIS(3),IOAXIS(4))                                 CHAXNM   28
      CALL ORIGIN (-5.,-3.)                                             15FEB    2
      CALL LBLAXES (IPLTPTR)                                            CHAXNM   29
      RETURN                                                            CHAXNM   30
      END                                                               CHAXNM   31
```

```
      SUBROUTINE CHTINM  (IPLTPTR)                                          CHTINM     2
C                                                                           CHTINM     3
C        THIS ROUTINE CHANGES THE TITLE NAME                                CHTINM     4
C                                                                           CHTINM     5
      COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),     /DISPLAY/   2
     X               IGAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),             6JULY77    2
     X               IDINSTR(19),ICDATA8(20,4,2),NCIP,IO8LOCK(20),NCIPM    1SEPT      1
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,          /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,          /PLTBEAD/   3
     X               IPFMN,IPFMV                                           /PLTBEAD/   4
      COMMON/DISKT/I8D1,ICOM1,ICOM2,XVAL,XX(20),YY(20),IC,NDEG              30AUG      1
C     J=M*2**19  DMSET(ICOM7+J,I90A,DATA)  STORES DATA(N) IN I8DA(1)        /DISKT/    3
      DIMENSION TITLE(4),TEXT(2)                                            CHTINM     9
      TEXT(1)=1HENTER TITL                                                  CHTINM    10
      TEXT(2)=1HE CR *                                                      CHTINM    11
      CALL TERASE (IDTITLE,4)                                               CHTINM    12
      A=ALPHIN(TEXT,20,TITLE,40,-600,700,NCH)                               CHTINM    13
      IF (A.NE.1H*) GO TO 50                                                CHTINM    14
C                                                                           CHTINM    15
C        NO TITLE--SET BEAD ADDRESS TO ZERO                                 CHTINM    16
      CALL DMSET (IPTITLE,IPLTPTR,0)                                        CHTINM    17
      RETURN                                                                CHTINM    18
C                                                                           CHTINM    19
C        TITLE--SET BEAD AND DISPLAY ON PLOT                                CHTINM    20
   50 CALL DMGTBD (10,ITPTR)                                                CHTINM    21
      CALL DMSET (ICOM1+2,ITPTR,NCH)                                        CHTINM    22
      CALL DMSET (ICOM1+6,IPLTPTR,ITPTR)                                    CHTINM    23
      NW=NCH/10                                                             CHTINM    24
      NR=NCH-10*NW                                                          CHTINM    25
      IF (NR.NE.0) NW=NW+1                                                  CHTINM    26
      JJ=NW*2**19+2                                                         CHTINM    27
      CALL DMSET (ICOM7+JJ,ITPTR,TITLE)                                     CHTINM    28
      CALL DISTITL (IPLTPTR)                                                CHTINM    29
      RETURN                                                                CHTINM    30
      END                                                                   CHTINM    31
```

```
      SUBROUTINE DATSCL(IPLTPTR)                                        DATSCL     2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J-N*2**16  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)    /DISKT/    3
      COMMON /TTEMP/TEMP(10)                                            /TTEMP/    2
      COMMON/TDIS/ITRIM1,FRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,      T30AUG     21
     1XDATMAX,YDATMIN,YDATMAX                                           T30AUG     22
      EXTERNAL SCCRV                                                    DATSCL     6
      DATA BIG/1.0E38/                                                  DATSCL     7
      XDATMIN=YDATMIN=BIG                                               20SEPT     35
      XDATMAX=YDATMAX=-BIG                                              20SEPT     36
      JJ=-4*2**12+4                                                     DATSCL     10
      CALL DMGET(ICOM1+8,IPLTPTR,ISC)                                   DATSCL     11
      IF(ISC.EQ.0)RETURN                                                DATSCL     12
      CALL DMGET(ICOM7+JJ,ISC,TEMP)                                     DATSCL     13
      IF(TEMP(1) .EQ.0..AND.TEMP(2).EQ.0.)GO TO 10                      DATSCL     14
      RETURN                                                            DATSCL     15
   10 CONTINUE                                                          DATSCL     16
      CALL DMGET(ICOM1+11,IPLTPTR,ICRVBD)                               DATSCL     17
      CALL SPLICE(ICRVBD,SCCRV,2,9,1,2,0)                               DATSCL     18
      TEMP(1)=XDATMIN                                                   DATSCL     19
      TEMP(2)=XDATMAX                                                   DATSCL     20
      TEMP(3)=YDATMIN                                                   DATSCL     21
      TEMP(4)=YDATMAX                                                   DATSCL     22
      CALL DMSET(ICOM7+JJ,ISC,TEMP)                                     DATSCL     23
      END                                                               DATSCL     24
```

```
      SUBROUTINE HARDCPY                                                    25AUG      7
      COMMON /PLOTCOM/ IFIRST,IPLTFL1,NCPC,ICALBD,DPLOT(128)                /PLOTCOM/  2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR     /BEADS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG              30AUG      1
C     J=M*2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)        /DISKT/    3
      NCPC=NCPC+1                                                           25AUG     14
      IF(NCPC.LE.100)GO TO 10                                               20SEPT    34
      CALL ERROFM(0.,5.,17HCALCOMP FILE FULL ,17,8HHARDCOPY,6HINFORM)       25AUG     16
      RETURN                                                                25AUG     17
   10 ICALBD=IPLTPTR                                                        LASTBUG    5
      CALL DMSET(ICOM1+10,IPLTPTR,99)                                       25AUG     19
   20 CALL DATSCL(IPLTPTR)                                                  1SEPT      6
      CALL AETSKC(5HPLOTC)                                                  14JUNE    41
      RETURN                                                                HARDCPY    5
      END                                                                   HARDCPY    6
```

```
      SUBROUTINE SCCRV(I9EAD1,I8EAD,IVALU)                              SCCRV     2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG     1
C     J=N*2**18  OMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/   3
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/  2
      COMMON /TDIS/A(8),XMN,XMX,YMN,YMX                                 SCCRV     5
      CALL OMGET(ICOM1+8,I8EAD,ICX)                                     SCCRV     6
      CALL OMGET(ICOM1+9,I9EAD,ICY)                                     SCCRV     7
      CALL OMGET(ICOM1+5,I8EAD,NNN)                                     SCCRV     8
      JJ=NNN*2**18                                                      SCCRV     9
      CALL OMGET(ICOM7+JJ,ICX,X)                                        SCCRV     10
      CALL OMGET(ICOM7+JJ,ICY,Y)                                        SCCRV     11
      QV=AMIN(X,NNN)                                                    SCCRV     12
      IF(QV.LT.XMN)XMN=QV                                               SCCRV     13
      QV=AMIN(Y,NNN)                                                    SCCRV     14
      IF(QV.LT.YMN)YMN=QV                                               SCCRV     15
      QV=AMAX(X,NNN)                                                    SCCRV     16
      IF(QV.GT.XMX)XMX=QV                                               SCCRV     17
      QV=AMAX(Y,NNN)                                                    SCCRV     18
      IF(QV.GT.YMX)YMX=QV                                               SCCRV     19
      END                                                               SCCRV     20
```

```
      SUBROUTINE TWINKLE (NN)                                          TWINKLE    2
C                                                                      TWINKLE    3
C        THIS ROUTINE BLINKS A STAR NEXT TO                            TWINKLE    4
C             1 - "HARDCOPY"                                           TWINKLE    5
C             2 - "PRINT PLOT INFORMATION"                             TWINKLE    6
C        AS A SIGNAL TO THE USER                                       TWINKLE    7
C                                                                      TWINKLE    8
C        ENTRY POINT EMBER ERASES THE STAR                             TWINKLE    9
C                                                                      TWINKLE   10
      COMMON/OPTION/IOPTION,ISHINE                                     LASTBUX    1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                 CYBTTT     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                6JULY77    3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                               /PARMSV/   2
      CALL ORIGIN (-1.,-4.)                                            TWINKLE   14
      XX=0.                                                            TWINKLE   15
      IF (IOPTION.EQ.4HTRIM) GO TO 30                                  TWINKLE   16
C                                                                      TWINKLE   17
C        CROSS                                                         TWINKLE   18
      YY=-1.45                                                         TWINKLE   19
      IF (NN.EQ.2) YY=-3.85                                            TWINKLE   20
      GO TO 50                                                         TWINKLE   21
C                                                                      TWINKLE   22
C        TRIM                                                          TWINKLE   23
   30 YY=-1.25                                                         TWINKLE   24
      IF (NN.EQ.2) YY=-5.25                                            TWINKLE   25
C                                                                      TWINKLE   26
C        DISPLAY STAR                                                  TWINKLE   27
   50 IXDGU=IFIX(XX*DGUPIN + XORDGU + .5)                              TWINKLE   28
      IYDGU=IFIX(YY*DGUPIN + YORDGU + .5)                              TWINKLE   29
      CALL GURSET (IXDGU,IYDGU,6,IBUF,NBYTE,MBYTE)                     TWINKLE   30
      CALL GUAN (1H*,1,IBUF,NBYTE,MBYTE)                               TWINKLE   31
      CALL GIDISP (NCON,IBUF,NBYTE,ISHINE)                             TWINKLE   32
C                                                                      TWINKLE   33
      RETURN                                                           TWINKLE   34
C        ENTRY POINT TO ERASE OLD STAR                                 TWINKLE   36
C                                                                      TWINKLE   37
      ENTRY EMBER                                                      TWINKLE   38
      IF (ISHINE.EQ.0) RETURN                                          TWINKLE   39
      CALL GIERAS (ISHINE)                                             TWINKLE   40
      RETURN                                                           TWINKLE   41
      END                                                              TWINKLE   42
```

```
      1                 SUBROUTINE SPLICE(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)                SPLICE     2
                    C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU       SPLICE     3
                    C     IN WORD JMATCH OF EACH BEAD.                                         SPLICE     4
                    C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND                SPLICE     5
      5             C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                       SPLICE     6
                    C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                   SPLICE     7
                    C     POINTER TO PREVIOUS BEAD IN WORD JPREV                               SPLICE     8
                    C     POINTER TO NEXT BEAD IN WORD JNEXT                                   SPLICE     9
                          DATAICOMP/01000000000008/                                            SPLICE    10
     10                   NEXT=IBD1                                                            SPLICE    11
                          IPREV=IBD1                                                           SPLICE    12
                          IF(IBD1.NE.0)GOTO10                                                  SPLICE    13
                          PRINT5                                                               SPLICE    14
                       5  FORMAT(13H IBD1 IS NULL )                                           SPLICE    15
     15                   IS=2                                                                 SPLICE    16
                          RETURN                                                               SPLICE    17
                      10  CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                               SPLICE    18
                          CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                  SPLICE    19
                          IF(ICKVALU.EQ.IVALU) GOTO15                                          SPLICE    20
     20                   IF(NEXTAD.EQ.0)GOTO12                                                SPLICE    21
                          IPREV=NEXT                                                           SPLICE    22
                          NEXT=NEXTAD                                                          SPLICE    23
                          GO TO 10                                                             SPLICE    24
                        12 WRITE (6,6) IVALU                                                   CYBTTT    21
     25                   6  FORMAT(19H NO MATCH FOR IVALU ,020)                               SPLICE    26
                          IS=2                                                                 SPLICE    27
                          RETURN                                                               SPLICE    28
                      15  CONTINUE                                                            SPLICE    29
                    C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL            SPLICE    30
     30                   IF(NEXT.EQ.IBD1)GO TO 25                                             SPLICE    31
                    C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE        SPLICE    32
                          IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                      SPLICE    33
                          IF(NEXTAD.EQ.0)GOTO22                                                SPLICE    34
                          CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                  SPLICE    35
     35                   IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                     SPLICE    36
                      22  CONTINUE                                                            SPLICE    37
                          CALL SUBR(IBD1,NEXT,IVALU)                                           SPLICE    38
                          IF(IS.EQ.1)CALL DMRLBD(NEXT)                                         SPLICE    39
                          RETURN                                                               SPLICE    40
     40                   25 CALL SUBR(IBD1,IBD1,IVALU)                                        SPLICE    41
                          IF(NEXTAD.EQ.0)GO TO 40                                              SPLICE    42
                          NEXT=NEXTAD                                                          SPLICE    43
                      30  CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                 SPLICE    44
                          CALL SUBR(IBD1,NEXT,IVALU)                                           SPLICE    45
     45                   IF(IS.EQ.1)CALL DMRLBD(NEXT)                                         SPLICE    46
                          IF(NEXTAD.EQ.0)GO TO 40                                              SPLICE    47
                          NEXT=NEXTAD                                                          SPLICE    48
                          GO TO 30                                                             SPLICE    49
                      40  CONTINUE                                                            SPLICE    50
     50                   IF(IS.EQ.1)CALL DMRLBD(IBD1)                                         SPLICE    51
                          END                                                                  SPLICE    52
```

```
1                FUNCTION AMIN(A,N)                                  AMIN      2
                 DIMENSION A(N)                                      AMIN      3
                 V=A(1)                                              AMIN      4
                 DO 10 I=2,N                                         AMIN      5
5                IF(A(I).LT.V)V=A(I)                                 AMIN      6
              10 CONTINUE                                            AMIN      7
                 AMIN=V                                              AMIN      8
                 END                                                 AMIN      9
```

```
1                FUNCTION AMAX(A,N)                                  AMAX      2
                 DIMENSION A(N)                                      AMAX      3
                 V=A(1)                                              AMAX      4
                 DO 10 I=2,N                                         AMAX      5
5                IF(A(I).GT.V) V=A(I)                                AMAX      6
            10   CONTINUE                                            AMAX      7
                 AMAX=V                                              AMAX      8
                 END                                                 AMAX      9
```

```
      SUBROUTINE VARINA (ITITLE,VALUE,N,IXOGU,IYOGU,IODAD)                VARINA     2
C                                                                         VARINA     3
C     ITITLE = ARRAY OF VARIABLE NAMES (10 CHAR WORDS - ONE NAME/WORD)    VARINA     4
C     VALUE  = ARRAY OF CURRENT PARAMETER VALUES                          VARINA     5
C     N      = NUMBER OF ITEMS TO BE DISPLAYED                            VARINA     6
C     IXOGU,IYOGU  =  UPPER LEFT CORNER OF TABLE (OGUS)                   VARINA     7
C     IODAD  = ARRAY TO TEMP. STORE ASSOCIATED ADDRESSES (DIMENSION N+1)  VARINA     8
C                                                                         VARINA     9
C     NAMED COMMON FOR GENERAL GRAPHICS CONSTANTS                         VARINA    10
C                                                                         VARINA    11
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                    CYB777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77    3
C                                                                         VARINA    13
C                                                                         VARINA    14
      DIMENSION ITITLE(1),VALUE(1),IODAD(1)                               VARINA    15
C                                                                         VARINA    16
      DATA KSAVE/1/                                                       VARINA    17
      KD=1                                                                VARINA    18
      NBYTE=0                                                             VARINA    19
      IY1=IXOGU                                                           VARINA    20
      IY2=IY1+312                                                         VARINA    21
      IX2=IX2+312                                                         VARINA    22
      DO 1 I=1,N                                                          VARINA    23
      IY1=IYOGU+75-75*I                                                   VARINA    24
      CALL GURSET(IX1,IY1,0033,IBUF,NBYTE,MBYTE)                          VARINA    25
      CALL GUAN(ITITLE(I),10,IBUF,NBYTE,MBYTE)                            VARINA    26
      CALL GURSET(IX2,IY1,1038,IBUF,NBYTE,MBYTE)                          VARINA    27
      CALL GUAN (VALUE(I),10,IBUF,NBYTE,MBYTE)                            VARINA    28
      KD=KD+1                                                             VARINA    29
    1 CALL GIDISP(NCON,IBUF,NBYTE,IODAD(KD),IBUT,KD,-0)                   VARINA    30
C                                                                         VARINA    31
      CALL GURSET(IXOGU+216,IYOGU+75-75*(N+1),1038,IBUF,NBYTE,MBYTE)      VARINA    32
      CALL GUAN(6HACCEPT,6,IBUF,NBYTE,MBYTE)                              VARINA    33
      KD=KD+1                                                             VARINA    34
      KEND=KD                                                             VARINA    35
      CALL GIDISP(NCON,IBUF,NBYTE,IODAD(KD),IBUT,KD,-0)                   VARINA    36
C                                                                         VARINA    37
    2 CALL GIBUT(0,NCON,IDDT,IODC,-0)                                     VARINA    38
      IF(IODC .GE. KSAVE+1 .AND. IODC .LE. KEND-1) GO TO 5                VARINA    39
      IF(IODC .EQ. KEND) GO TO 3                                          VARINA    40
      GO TO 2                                                             VARINA    41
    3 NE=KEND-KSAVE                                                       VARINA    42
      DO 4 I=1,NE                                                         VARINA    43
    4 CALL GIERAS(IODAD(KSAVE+I),-0)                                      VARINA    44
      KD=KSAVE                                                            VARINA    45
      RETURN                                                              VARINA    46
C                                                                         VARINA    47
    5 ICHANG=IODC                                                         VARINA    48
      IY2=IYOGU+75-75*(IODC-KSAVE)                                        VARINA    49
    6 NL=10                                                               VARINA    50
      CALL GIEOM (NCON,IEOM,IBUT,KEND+1,-0)                               VARINA    51
      CALL GIANS(NCON,NL,IX3,IY2)                                         VARINA    52
C                                                                         VARINA    53
    7 CALL GIBUT(0,NCON,IDDT,IODC,-0)                                     VARINA    54
      IF(IODC .EQ. KEND+1) GO TO 8                                        VARINA    55
      GO TO 7                                                             VARINA    56
C                                                                         VARINA    57
```

```
      8 CALL GIANE(NCON,NL,REG)                                          VARINA    58
        IF(NL .LT. 1) GO TO 6                                            VARINA    59
60      II=ICHANG-KSAVE                                                  VARINA    60
        VALUE(II)=REG                                                    VARINA    61
      C                                                                  VARINA    62
        CALL GIERAS(IODAD(ICHANG),-0)                                    VARINA    63
      C                                                                  VARINA    64
65      CALL GURSET(IX1,IY2,1039,IBUF,NBYTE,MBYTE)                       VARINA    65
        CALL GUAN(ITITLE(II),10,IBUF,NBYTE,MBYTE)                        VARINA    66
        CALL GURSET(IX2,IY2,1039,IBUF,NBYTE,MBYTE)                       VARINA    67
        CALL GUAN (VALUE(II),10,IBUF,NBYTE,MBYTE)                        VARINA    68
        CALL GIDISP(NCON,IBUF,NBYTE,IODAD(ICHANG),IBUT,ICHANG,-0)        VARINA    69
70    C                                                                  VARINA    70
        GO TO 2                                                          VARINA    71
      C                                                                  VARINA    72
        END                                                              VARINA    73
```

```
      SUBROUTINE DISTITL(IPLTPTR)                                        14JUNE    11
C                                                                        DISTITL    3
C         THIS ROUTINE DISPLAYS THE TITLE AT THE TOP OF THE              DISTITL    4
C         PLOT. THE NUMBER OF LINES IS LIMITED TO THREE.                 DISTITL    5
C                                                                        DISTITL    6
      COMMON/COMPCOD/IWDZERO,IWDONE,IWDTWO,IWDTHRE,IWDFOUR,IWDFIVE,      /COMPCOD/   2
     X               IWDSIX,IWDSEVE,IWDEIGH,IWDNINE,IWDTEN               /COMPCOD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77     2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT       1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYB777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77     3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/   3
     X               IPFMN,IPFMV                                        /PLTBEAD/   4
      INTEGER STRING(8)                                                  DISTITL   11
      CALL ORIGIN (-5.,-3.)                                              14JUNE    13
      Y=6.0                                                              DISTITL   12
      LINECNT=0                                                          DISTITL   13
C                                                                        DISTITL   14
C         GET THE TITLE TEXT POINTER FROM THE PLOT GROUP BEAD. IF THE    DISTITL   15
C         POINTER IS ZERO, THEN THERE IS NO TITLE TO DISPLAY.            DISTITL   16
C                                                                        DISTITL   17
      CALL DMGET (IPTITLE,IPLTPTR,ITPTR)                                 14JUNE    14
   10 IF(ITPTR .EQ. 0) GO TO 20                                          DISTITL   19
C                                                                        DISTITL   20
C         THERE IS A LINE OF TITLE TO BE DISPLAYED. GET THE NUMBER       DISTITL   21
C         OF CHARACTERS AND THE TITLE TEXT FROM THE TITLE TEXT BEAD      DISTITL   22
C         AND DISPLAY THE LINE.                                          DISTITL   23
C                                                                        DISTITL   24
      CALL DMGET(IWDTWO,ITPTR,NCHAR)                                     DISTITL   25
      CALL DMGET(700100000038,ITPTR,STRING)                              DISTITL   26
      Y=Y-.25                                                            DISTITL   27
      LINECNT = LINECNT + 1                                              DISTITL   28
      CALL ANNOTA(.42,Y,STRING,NCHAR,IDTITLE(LINECNT),IGN,-0,-0,-0)      DISTITL   29
C                                                                        DISTITL   30
C         IF THE LINE LIMIT HAS NOT BEEN MET, GET THE POINTER TO THE     DISTITL   31
C         NEXT TITLE TEXT BEAD. OTHERWISE, RECORD THE NUMBER OF LINES    DISTITL   32
C         WHICH WERE DISPLAYED.                                          DISTITL   33
C                                                                        DISTITL   34
      IF(LINECNT .EQ. 3) GO TO 20                                        DISTITL   35
      CALL DMGET(IWDONE,ITPTR,ITPTR)                                     DISTITL   36
      GO TO 10                                                           DISTITL   37
   20 IDTITLE(4) = LINECNT                                               DISTITL   38
      RETURN                                                             DISTITL   39
      END                                                                DISTITL   40
```

```
      SUBROUTINE LBLAXES(IPLTPTR)                                        LBLAXES    2
C                                                                        LBLAXES    3
C         THIS ROUTINE LABELS THE X AND Y AXIS OF A PREVIOUSLY DISPLAYED LBLAXES    4
C         AXIS SYSTEM. PRIMARY INPUT TO THE ROUTINE IS THE PLOT BEAD     LBLAXES    5
C         POINTER.                                                       LBLAXES    6
C                                                                        LBLAXES    7
      COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1, /AXBEAD/   2
     X              IAXY2,IAXY3                                         /AXBEAD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(30),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPN 1SEPT      1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYBER77    1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77    3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/   3
     X               IPFMN,IPFMV                                        /PLTBEAD/   4
C                                                                        LBLAXES   12
C         GET THE AXIS NAMES FROM THE AXIS BEAD.                         LBLAXES   13
C                                                                        LBLAXES   14
      CALL DMGET(IPAXIS,IPLTPTR,IAXPTR)                                  LBLAXES   15
      CALL DMGET(IAXXNME,IAXPTR,IXNAME)                                  LBLAXES   16
      CALL DMGET(IAXY1,IAXPTR,IYNAME)                                    LBLAXES   17
C                                                                        LBLAXES   18
C         DISPLAY BOTH AXIS NAMES.                                       LBLAXES   19
C                                                                        LBLAXES   20
      CALL ANNOTA (4.02,-.8,IXNAME,10,IDAXIS(3),IGN,-0,-0,-0)            20SEPT     6
      CALL ANNOTA (-.36,6.37,IYNAME,10,IDAXIS(4),IGN,-0,-0,-0)           20SEPT     7
      RETURN                                                             LBLAXES   23
      END                                                                LBLAXES   24
```

```
1                SUBROUTINE RESCALE(IPLTPTR)                                    RESCALE     2
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG       30AUG       1
           C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/     3
                 DIMENSION A(8)                                                 RESCALE     4
5                DATA A/4HXMIN, 7HDELTA X, 4HYMIN,7HDELTA Y,9HDATA XMIN,9HDATA XMAX RESCALE     5
                1,9HDATA YMIN,9HDATA YMAX/                                      RESCALE     6
                 CALL DMGET(ICOM1+8,IPLTPTR,ISCPTR)                             RESCALE     7
                 JJ=8*2**18                                                     30AUG       2
           C     JJ=4*2**18                                                     30AUG       3
10               CALL DMGET(ICOM7+JJ,ISCPTR,XX)                                 RESCALE    10
                 CALL VARIN (A,XX,8,-1800,1000,YY)                              30AUG       4
           C     CALL VARIN (A,XX,4,-1000,1000,YY)                              30AUG       5
                 CALL DMSET(ICOM7+JJ,ISCPTR,XX)                                 T30AUG     25
                 END                                                            RESCALE    13
```

```
      SUBROUTINE VARIN(ITITLE,VALUE,N,IXDGU,IYDGU,IDDAD)                    VARIN      2
C                                                                           VARIN      3
C     ITITLE = ARRAY OF VARIABLE NAMES (10 CHAR WORDS - ONE NAME/WORD)      VARIN      4
C     VALUE  = ARRAY OF CURRENT PARAMETER VALUES                            VARIN      5
C     N      = NUMBER OF ITEMS TO BE DISPLAYED                              VARIN      6
C     IXDGU,IYDGU  =  UPPER LEFT CORNER OF TABLE (DGUS)                     VARIN      7
C     IDDAD  = ARRAY TO TEMP. STORE ASSOCIATED ADDRESSES (DIMENSION N+1)    VARIN      8
C                                                                           VARIN      9
C     NAMED COMMON FOR GENERAL GRAPHICS CONSTANTS                           VARIN     10
C                                                                           VARIN     11
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                      CYB777     1
     >    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                      6JULY77    3
C                                                                           VARIN     13
C                                                                           VARIN     15
      DIMENSION ITITLE(1),VALUE(1),IDDAD(1)                                 VARIN     16
C                                                                           VARIN     17
      DATA KSAVE/1/                                                         26AUG     17
      KD=1                                                                  VARIN     19
      NBYTE=0                                                               VARIN     20
      IX1=IXDGU                                                             VARIN     21
      IY2=IX1+312                                                           VARIN     22
      IX2=IX2+312                                                           VARIN     23
      DO 1 I=1,N                                                            VARIN     24
      ENCODE(10,100,IDUM) VALUE(I)                                          VARIN     25
  100 FORMAT(G10.3)                                                         VARIN     26
      IY1=IYDGU+75-75*I                                                     VARIN     27
      CALL GURSET(IX1,IY1,0038,IBUF,NBYTE,MBYTE)                            VARIN     28
      CALL GUAN(ITITLE(I),10,IBUF,NBYTE,MBYTE)                              VARIN     29
      CALL GURSET(IX2,IY1,1038,IBUF,NBYTE,MBYTE)                            VARIN     30
      CALL GUAN(IDUM,10,IBUF,NBYTE,MBYTE)                                   VARIN     31
      KD=KD+1                                                               VARIN     32
    1 CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(KD),IBUT,KD,-0)                     VARIN     33
C                                                                           VARIN     34
      CALL GURSET(IXDGU+216,IYDGU+75-75*(N+1),1038,IBUF,NBYTE,MBYTE)        VARIN     35
      CALL GUAN(6HACCEPT,6,IBUF,NBYTE,MBYTE)                                VARIN     36
      KD=KD+1                                                               VARIN     37
      KEND=KD                                                               VARIN     38
      CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(KD),IBUT,KD,-0)                     VARIN     39
C                                                                           VARIN     40
    2 CALL GIBUT(0,NCON,IDDT,IDDC,-0)                                       VARIN     41
      IF(IDDC .GE. KSAVE+1 .AND. IDDC .LE. KEND-1) GO TO 5                  VARIN     42
      IF(IDDC .EQ. KEND) GO TO 3                                            VARIN     43
      GO TO 2                                                               VARIN     44
    3 NE=KEND-KSAVE                                                         VARIN     45
      DO 4 I=1,NE                                                           VARIN     46
    4 CALL GIERAS(IDDAD(KSAVE+I),-0)                                        VARIN     47
      KD=KSAVE                                                              VARIN     48
      RETURN                                                                VARIN     49
C                                                                           VARIN     50
    5 ICHANG=IDDC                                                           VARIN     51
      IY2=IYDGU+75-75*(IDDC-KSAVE)                                          VARIN     52
    6 NL=10                                                                 VARIN     53
      CALL GIEOM (NCON,IEOM,IBUT,KEND+1,-0)                                 CYB777    25
      CALL GIANS(NCON,NL,IX3,IY2)                                           VARIN     55
C                                                                           VARIN     56
    7 CALL GIBUT(0,NCON,IDDT,IDDC,-0)                                       VARIN     57
      IF(IDDC .EQ. KEND+1) GO TO 8                                          VARIN     58
```

```
                GO TO 7                                                  VARIN      59
              C                                                          VARIN      60
   60         8 CALL GIANI(NCON,NL,LREG)                                 VARIN      61
                IF(NL .LT. 1) GO TO 6                                    VARIN      63
                ENCODE (7,128,IFMT) NL                                   VARIN      64
             128 FORMAT(2H(5,I2,3H.0))                                   VARIN      65
                IT=ICHANG-KSAVE                                          VARIN      66
   65           DECODE(NL,IFMT,LREG) VALUE(II)                           VARIN      67
                CALL EROK,RETURNS(6)                                     5OCT      107
              C                                                          VARIN      71
                29 ENCODE(10,100,IDUM) VALUE(II)                         VARIN      72
                CALL GIEPS(IDDAD(ICHANG),-0)                             VARIN      73
   70         C                                                          VARIN      74
                CALL GURSET(IX1,IY2,0033,IBUF,NBYTE,MBYTE)               VARIN      75
                CALL GUAN(ITITLE(II),10,IBUF,NBYTE,MBYTE)                VARIN      76
                CALL GURSET(IX2,IY2,1033,IBUF,NBYTE,MBYTE)               VARIN      77
                CALL GUAN(IDUM,10,IBUF,NBYTE,MBYTE)                      VARIN      78
   75           CALL GIOTSP(NCON,IBUF,NBYTE,IDDAD(ICHANG),IBUT,ICHANG,-0) VARIN     79
              C                                                          VARIN      80
                GO TO 2                                                  VARIN      81
              C                                                          VARIN      82
                END                                                      VARIN      83
```

```
      SUBROUTINE WINDEX                                                 WINDEX      2
C                                                                       WINDEX      3
C         THIS ROUTINE CLEARS THE SCREEN, RESETS THE SYSTEM             WINDEX      4
C         MASK VALUES, AND REDISPLAYS THE STOP BUTTON.                  WINDEX      5
C                                                                       WINDEX      6
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYB777      1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77     3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),        6JULY77     2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
      CALL TERASE (IDMENU,IDMENU(20))                                   25JULY77    12
      CALL GICNJB(NCON)                                                 WINDEX      7
      CALL GIMASK(NCON,0,IGN,IGN)                                       WINDEX      9
      CALL GIMASK(NCON,0,ISIN,ISIN)                                     14APR76    33
      CALL GIMASK(NCON,0,ISTR,ISTR)                                     14APR76    34
      CALL GIMASK(NCON,0,IBUT,IBUT)                                     14APR76    35
      CALL GIMASK(NCON,0,ISIN + ISTR + IBUT,IMARK)                      14APR76    36
C                                                                       WINDEX     14
      CALL ORIGIN(0.,0.)                                                WINDEX     15
      CALL ANNOTA(-.24,9.5,4HSTOP,4,IDSTOP,IBUT,254,-0,-0)              14APR76    37
      CALL GIBUT(1,NCON,IDDT)                                           WINDEX     17
      RETURN                                                            WINDEX     18
      END                                                               WINDEX     19
```

```
      1                 OVERLAY (4,6)                                          MENU        2
                        PROGRAM MENU                                           MENU        3
                  C                                                            MENU        4
                  C       CALL APPROPRIATE MENU ROUTINE                        MENU        5
      5           C                                                            MENU        6
                        COMMON/OPTION/IOPTION,ISHINE                           LASTBUX     1
                        IF (IOPTION .EQ. 10HCROSS PLOT) CALL CPMENU            MENU        8
                        IF (IOPTION .EQ. 4HTRIM) CALL TMENU                    MENU        9
                        CALL AEPTRN                                            MENU       10
     10                 END                                                    MENU       11
```

```
      SUBROUTINE CPMENU                                                 CPMENU      2
C                                                                       CPMENU      3
C         THIS ROUTINE DISPLAYS THE CROSS PLOT MENU.                    CPMENU      4
C                                                                       CPMENU      5
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNENO(10),IOSPEC(19),IOCF(41), /DISPLAY/  2
     X               IDAXIS(4),IOTITLE(4),IOCO(23),IDMENU(20),        6JULY77     2
     X               IDINSTP(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT     1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBER77     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77     3
      CALL ORIGIN(-1.,-4.)                                              CPMENU      8
      SY=-.25                                                           10MAY77    18
      SDY=.4                                                            14JUNE     27
      SY=.2                                                             10MAY77    20
      CALL ANNOTA (SX,SY,4HEDIT,4,IDMENU(1),IBUT,1,-0,-0)               10MAY77    21
      SY=SY-SDY                                                         10MAY77    22
      CALL ANNOTA (SX,SY,9HCURVE FIT,9,IDMENU(2),IBUT,2,-0,-0)          10MAY77    23
      SY=SY-SDY                                                         10MAY77    24
      CALL ANNOTA (SX,SY,23HPREFIT REMAINING CURVES,23,IDMENU(3),       10MAY77    25
     1    IBUT,13,-0,-0)                                              10MAY77    26
      SY=SY-SDY                                                         10MAY77    27
      CALL ANNOTA (SX,SY,8HHARDCOPY,8,IDMENU(4),IBUT,3,-0,-0)           10MAY77    28
      SY=SY-SDY                                                         10MAY77    29
      CALL ANNOTA (SX,SY,9HNEXT PLOT,9,IDMENU(5),IBUT,4,-0,-0)          10MAY77    30
      SY=SY-SDY                                                         10MAY77    31
      CALL ANNOTA (SX,SY,7HANALYZE,7,IDMENU(6),IBUT,5,-0,-0)            10MAY77    32
      SY=SY-SDY                                                         10MAY77    33
      CALL ANNOTA (SX,SY,12HCHANGE SCALE,12,IDMENU(7),IBUT,10,-0,-0)    10MAY77    34
      SY=SY-SDY                                                         10MAY77    35
      CALL ANNOTA (SX,SY,17HCHANGE AXIS NAMES,17,IDMENU(8),IBUT,        10MAY77    36
     1    11,-0,-0)                                                   10MAY77    37
      SY=SY-SDY                                                         14JUNE     28
      CALL ANNOTA (SX,SY,12HCHANGE TITLE,12,IDMENU(12),IBUT,14,-0,-0)   14JUNE     29
      SY=SY-SDY                                                         14JUNE     30
      CALL ANNOTA (SX,SY,22HPRINT PLOT INFORMATION,22,IDMENU(13),       14JUNE     31
     1    IBUT,15,-0,-0)                                              14JUNE     32
      SY=SY-SDY                                                         10MAY77    38
      CALL ANNOTA (SX,SY,9HPLOT TRIM,9,IDMENU(9),IBUT,8,-0,-0)          10MAY77    39
      SY=SY-SDY                                                         10MAY77    40
      CALL ANNOTA (SX,SY,14HCROSS PLOT TWO,14,IDMENU(10),IBUT,12,-0,-0) 10MAY77    41
      SY=SY-SDY                                                         10MAY77    42
      CALL ANNOTA (SX,SY,35HRETURN TO PLOT GROUP SPECIFICATIONS,        10MAY77    43
     1    35,IDMENU(11),IBUT,6,-0,-0)                                 10MAY77    44
      IDMENU(20)=13                                                     14JUNE     33
      RETURN                                                            CPMENU     17
      END                                                               CPMENU     18
```

```
      SUBROUTINE TMENU                                                  TMENU    2
C                                                                       TMENU    3
C         THIS ROUTINE DISPLAYS THE TRIM MENU.                          TMENU    4
C                                                                       TMENU    5
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/ 2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),        6JULY77  2
     X               IDINSTR(19),ICDATA8(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT  1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY5777   1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77  3
      CALL ORIGIN(-1.,-4.)                                              TMENU    8
      CALL ANNOTA(.2,-.25,4HEDIT,4,IDMENU(1),IBUT,1,-0,-0)              TMENU    9
      CALL ANNOTA(.2,-.75,9HCURVE FIT,9,IDMENU(2),IBUT,2,-0,-0)         TMENU   10
      CALL ANNOTA(.2,-1.25,8HHARDCOPY,8,IDMENU(3),IBUT,3,-0,-0)         TMENU   11
      CALL ANNOTA(.2,-1.75,9HNEXT PLOT,9,IDMENU(4),IBUT,4,-0,-0)        TMENU   12
      CALL ANNOTA(.2,-2.25,8HNEXT SET,8,IDMENU(5),IBUT,7,-0,-0)         TMENU   13
      CALL ANNOTA (.2,-2.75,9HPLOT TRIM,9,IDMENU(6),IBUT,8,-0,-0)       5OCT    31
      CALL ANNOTA(.2,-3.25,12HCOMPUTE TRIM,12,IDMENU(7),IBUT,9,-0,-0)   5OCT    32
      CALL ANNOTA (.2,-3.75,12HCHANGE SCALE,12,IDMENU(8),IBUT,10,-0,-0) 26AUG   12
      CALL ANNOTA (.2,-4.25,17HCHANGE AXIS NAMES,17,                    AXNME    10
     1    IDMENU(9),IBUT,11,-0,-0)                                      AXNME    11
      CALL ANNOTA (.2,-4.75,12HCHANGE TITLE,12,IDMENU(10),IBUT,14,-0,-0) 14JUNE  59
      CALL ANNOTA (.2,-5.25,22HPRINT PLOT INFORMATION,22,               14JUNE   60
     1    IDMENU(13),IBUT,15,-0,-0)                                     14JUNE   61
      CALL ANNOTA(.2,-5.75,23HPREFIT REMAINING CURVES, 23,IDMENU(12),   14JUNE   62
     1IBUT,13,-0,-0)                                                    14JUNE   63
      CALL ANNOTA (.2,-6.25,35HRETURN TO PLOT GROUP SPECIFICATIONS,     14JUNE   64
     1    35,IDMENU(11),IBUT,6,-0,-0)                                   14JUNE
      IDMENU(20)=13                                                     14JUNE
      RETURN                                                            TMENU    19
      END                                                               TMENU    20
```

```
      OVERLAY (4,4)                                                          FIXINH    17
      PROGRAM CROSSX                                                         FIXINH    18
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                       CY8777     1
     X     ISTP,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                      6JULY77    3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1         10MAY77   61
     1,IDELTE,IDELBD(2)                                                      FLCAD2     5
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)      /CROSTB/   2
     1,EMACH,PVALU(10),PNAME(10),IRP(10)                                     /CROSTB/   3
      COMMON /MULTKS/ NKS,AXK(20)                                            /MULTKS/   2
      COMMON/IBADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR      /BEADS/    2
      COMMON/SLFIT/ICB12,ICHNG                                               /SLFIT/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG      1
C     J=N*2**13  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/    3
      CALL ORIGIN (-3.,-4.)                                                  CROSSX     8
      CALL ANNOTA(0.,0.,32H PICK CROSSPLOT TYPE,THEN ACCEPT,32,PNAME(6),     CROSSX     9
     1IGN,6,-3,-0)                                                           CROSSX    10
      YOFF=.5                                                                CROSSX    11
      XV=1.0                                                                 CROSSX    12
      YV=0.                                                                  CROSSX    13
      DO 10 I=1,4                                                            CROSSX    14
      YV=YV-YOFF                                                             CROSSX    15
   10 CALL ANNOTA(XV,YV,CTYPE(I),10,PNAME(I),ISIN,I,-0,-0)                   CROSSX    16
      YV=YV-1.                                                               CROSSX    17
      CALL ANNOTA(XV,YV,6HACCEPT,6,PNAME(5),IBUT,5,-0,-0)                    CROSSX    18
   11 CALL GIBUTI(0,NCON,IDOTR,IDOCR)                                        CROSSX    19
      IF(IDOCR.NE.5)GO TO 11                                                 CROSSX    20
      CALL GIFID (NCON,IDOTR,IDOCR)                                          CROSSX    21
      IF (IDOTR.NE.0) GO TO 12                                               CROSSX    22
      CALL ERROFM(XV,YV-1.,35HNO SINGLE PICK, FOLLOW INSTRUCTIONS,35,        CROSSX    23
     15HCROSS,6HINFORM)                                                      CROSSX    24
      GO TO 11                                                               CROSSX    25
   12 JT=IDOCR                                                               CROSSX    26
      CALL TERASE(PNAME,6)                                                   CROSSX    27
C     ENTER TYPE,INDEP VAR,Y-NAME,                                           CROSSX    28
      YNAME=TYPE=CTYPE(JT)                                                   CROSSX    29
      CALL ANPLGR(IPLTPTR,1,3,IONE)                                          CROSSX    30
      IF(ICPE.EQ.0)GO TO 1201                                                CROSSX    31
      CALL ERROFM(XV,YV-1.,28HSINGLE PLOT IN GROUP ,20,5HCROSS,6HINFORM)     CROSSX    32
      CALL AERTRN                                                            LASTBUX    4
 1201 CONTINUE                                                               CROSSX    34
C     IF TYPE .NE. MINIMUM ENTER K                                           CROSSX    35
      IF(JT.EQ.3.OR.JT.EQ.5)GO TO 14                                         FIXMIN     1
      IRP(1)=10HENTER VALU                                                   14JUNE    34
      IRP(2)=10HE OF INDEP                                                   14JUNE    35
      IRP(3)=10HENDENT VAR                                                   14JUNE    36
      IRP(4)=10HIABLE FOR                                                    14JUNE    37
      IRP(5)=10HANALYZE                                                      14JUNE    38
    5 CALL ENTER(IRP,50,0.,0.,AXK,NKS)                                       14JUNE    39
      IF(NKS.GT.1)NKS=1                                                      CROSSX    42
      IF(NKS.LT.1)GO TO 5                                                    CROSSX    43
   14 CONTINUE                                                               CROSSX    44
      IF (IPLTPTR.EQ.0) GO TO 100                                            CROSSX    45
      CALL WINDEX                                                            CROSSX    46
      ICB12=3                                                                SLOPES     6
      IF(JT.NE.2)GO TO 99                                                    SLOPES     7
      CALL DMGET(ICOM1+11,IPLTPTR,ICVBD)                                     SLOPES     8
      CALL DMGET(ICOM1+11,ICVBD,IFITBD)                                      SLOPES     9
```

```
      CALL CMGET(ICCM1+1,IFITBD,ITYPE)                                  SLOPES    10
      IF(ITYPE.NE.6HAITKEN)GO TO 99                                     SLOPES    11
60    ICHNG=1                                                           SLOPES    12
      ICF12=1                                                           SLOPES    13
      CALL AFTSKC(5HDOVER)                                              SLOPES    14
99    CONTINUE                                                          SLOPES    15
      CALL AFTSKC (5HCROSS)                                             CROSSX    47
65    100 CALL ERRORM(XV,YV-1.,17HNULL PLOT POINTER ,17,5HCROSS,6HINFORM) CROSSX    48
      CALL AFRTRN                                                       LASTBUX    5
      END                                                               CROSSX    50
```

```
      SUBROUTINE RWPLGR(IBEAD1,INEXT,IPREV,IONE)                        RWPLGR     2
      COMMON/BEADS/IPGPTR,IAXPTR,IAMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**19  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      EXTERNAL SUBGET                                                   RWPLGR     5
C     IONE=1 MEANS ONLY ONE PLOT IN GROUP                               RWPLGR     6
      IONE=0                                                            RWPLGR     7
      CALL DMGET(ICOM1+IPREV,IBEAD1,IPR)                                RWPLGR     8
      IF(IPR.NE.0)GO TO 10                                              RWPLGR     9
   10 CALL DMGET(ICOM1+INEXT,IBEAD1,NEXT)                               RWPLGR    10
      IF(NEXT.NE.0)RETURN                                               RWPLGR    11
      IONE=1                                                            RWPLGR    12
      RETURN                                                            RWPLGR    13
   15 CALL SPLICE(IBEAD1,SUBGET,IPREV,0,IPREV,INEXT,0)                  RWPLGR    14
      END                                                               RWPLGR    15
```

```
1               SUBROUTINE SUBGET(IBEAD1,IBEAD,IVALU)                    RMPLGR     16
                IBEAD1=IBEAD                                             RMPLGR     17
                END                                                      RMPLGR     18
```

```
      SUBROUTINE ENTER(TEXT,NT,XV,YV,RESULTS,NRES)                      ENTER      2
C     DISPLAYS A LINE OF NT CHARACTERS FROM ARRAY TEXT AT LOCATION (XV,Y ENTER      3
C     A LIGHT REGISTER PERMITS INPUTTING A STRING OF UP TO 80 CHARACTERS ENTER      4
C     INPUT IS NUMERIC DATA SEPARATED BY COMMAS                         ENTER      5
C     RETURNED IS NC, THE NUMBER OF CHARACTERS ENTERED                  ENTER      6
C     DECODED NUMERIC VALUES ARE RETURNED THROUGH THE ARRAY RESULTS     ENTER      7
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYB777     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                /PARMSV/   2
      DIMENSION TEXT(1),RESULTS(1),DELIM(3),BCD(8)                      ENTER      9
      DATA DELIM/1L&,1L ,1L,/,NDL/3/                                    ENTER     10
      CALL ANNOTA(XV,YV,TEXT,NT,IDAD,1,99,-0,-0)                        ENTER     11
      IH=XV*DGUPIN+XORDGU                                               10MAY77   72
      IV=YV*DGUPIN+YORDGU-50.                                           10MAY77   73
      CALL GIEOM (NCON,IEOM,IBUT,200)                                   CYB777    30
  298 NC=40                                                             10MAY77   74
  299 CALL GIAMS(NCON,NC,IH,IV)                                         ENTER     15
  300 CALL GIBUT(0,NCON,TDDT,IDDC)                                      ENTER     16
      IF(IDDC.NE.200)GO TO 300                                          ENTER     17
      CALL GIANE(NCON,NC,BCD)                                           ENTER     18
      CALL STRING (1,DELIM,1,BCD,NC+1)                                  CYB777    32
      CALL SELECT(BCD,8,DELIM,NDL,RESULTS,NRES)                         ENTER     19
      CALL ERCK,RETURNS(298)                                            ENTER     20
      CALL GIERAS(IC40)                                                 ENTER     21
      END                                                               ENTER     22
```

```
      SUBROUTINE SELECT(BCD,NW,DELIM,NDL,RESULTS,NRES)                  SELECT     2
C FOR INFORMATION CONTACT T. ROWLAND  4950/ADDS  56247                  SELECT     3
C                                                                       SELECT     4
C     DIMENSION BCD(NW),DELIM(NDL),RESULTS(NRES)                        SELECT     5
C                                                                       SELECT     6
C                                                                       SELECT     7
C     SELECT DECODES A STRING OF FREE FIELD INFORMATION WHERE           SELECT     8
C                                                                       SELECT     9
C          BCD IS INPUT ARRAY OF LENGTH NW CONTAINING A STRING         SELECT    10
C        OF FREE FIELD INFORMATION.                                     SELECT    11
C                                                                       SELECT    12
C          DELIM IS INPUT ARRAY OF LENGTH NDL CONTAINING LEFT          SELECT    13
C        JUSTIFIED, ZERO-FILLED DELIMITER CHARACTERS.                   SELECT    14
C                                                                       SELECT    15
C          DELIM(1) MUST CONTAIN THE DELIMITER USED AS AN              SELECT    16
C        END-OF-STRING INDICATOR.                                       SELECT    17
C                                                                       SELECT    18
C          RESULTS IS A RETURNED ARRAY OF LENGTH NRES CONTAINING       SELECT    19
C        DECODED FLOATING-POINT NUMBERS.                                SELECT    20
C                                                                       SELECT    21
C     CALLS SUBROUTINE GETCH(GET I-TH CHARACTER OF J-TH WORD)           SELECT    22
C     AND   SUBROUTINE PUTCH (PUT CHARACTER IN I-TH POSITION OF J-TH WOR SELECT    23
C                                                                       SELECT    24
C                                                                       SELECT    25
C                                                                       SELECT    26
      DIMENSION BCD(1),DELIM(1),RESULTS(1),TEMP(2)                      SELECT    27
      INTEGER C,DELIM                                                   SELECT
      TEMP(1)=TEMP(2)=0                                                 SELECT
      NC=0                                                              SELECT    30
      II=JJ=K=1                                                         SELECT    31
      DO 90 J=1,NW                                                      SELECT    32
      DO 90 I=1,10                                                      SELECT    33
      CALL GETCH(C,BCD,I,J)                                             SELECT    34
      DO 5 LLL=1,NDL                                                    SELECT    35
      IF(C.EQ.DELIM(LLL))GO TO 20                                       SELECT    36
    5 CONTINUE                                                          SELECT    37
      NC=NC+1                                                           SELECT    38
      CALL PUTCH(C,TEMP,II,JJ)                                          SELECT    39
      II=II+1                                                           SELECT    40
      IF(II.LE.10)GO TO 90                                              SELECT    41
      II=1                                                              SELECT    42
      JJ=2                                                              SELECT    43
      GO TO 90                                                          SELECT    44
   20 IF(NC.EQ.0)GO TO 90                                               SELECT    45
      ENCODE(10,7,FMT)NC                                                SELECT    46
    7 FORMAT(2H(E,I2,3H.0))                                             SELECT    47
      DECODE(NC,FMT,TEMP)RESULTS(K)                                     SELECT    48
      K=K+1                                                             SELECT    49
      IF(C.NE.DELIM(1))GO TO 30                                         SELECT    50
   25 NRES=K-1                                                          SELECT    51
      RETURN                                                            SELECT    52
   30 II=1                                                              SELECT    53
      JJ=1                                                              SELECT    54
      TEMP(1)=TEMP(2)=0                                                 SELECT    55
      NC=0                                                              SELECT    56
   90 CONTINUE                                                          SELECT    57
      NRES=K-1                                                          SELECT    58
```

```
      RETURN
      END
```

SELECT 59
SELECT 60

```
      SUBROUTINE GETCH(L,W,I,J)                                          GETCH      2
C                                                                        GETCH      3
C     GETCH RETURNS THE I-TH CHARACTER OF WORD W(J) LEFT JUSTIFIED       GETCH      4
C        IN LOCATION L WITH ZERO FILL.                                   GETCH      5
C                                                                        GETCH      6
      DIMENSION W(J)                                                     GETCH      7
      DATA MSK/77000000000000000009/                                     GETCH      8
      K=6*(I-1)                                                          GETCH      9
      L=MSK.AND.SHIFT(W(J),K)                                            GETCH     10
      END                                                                GETCH     11
```

```
      SUBROUTINE PUTCH(CH,W,I,J)                                        PUTCH      2
C                                                                       PUTCH      3
C     PUTS CHARACTER CH (LEFT JUSTIFIED) IN POSITION I OF WORD(J)       PUTCH      4
C                                                                       PUTCH      5
C     CAUTION, I-TH POSITION OF J-TH WORD MUST CONTAIN BINARY ZEROS.    PUTCH      6
C                                                                       PUTCH      7
      DIMENSION W(J)                                                    PUTCH      8
      DATA MSK/77000000000000000000B/                                   PUTCH      9
      K=6*(11-I)                                                        PUTCH     10
      W(J)=W(J).OR.SHIFT(CH.AND.MSK,K)                                  PUTCH     11
      END                                                               PUTCH     12
```

```
      1                OVERLAY (4,5)                                                    FIXINM     19
                       PROGRAM CROSS2X                                                  FIXINM     20
                       COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                 CVB777      1
                      X     IST*,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO               6JULY77     3
      5                COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1   10MAY77     61
                      1,IDELTF,IDELBD(2)                                               FLCABD      5
                       COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10) /CROSTB/    2
                      1,FRACH,PVALU(10),PNAME(10),IRP(10)                              /CROSTB/    3
                       COMMON /MULTKS/ NKS,AXK(20)                                     /MULTKS/    2
     10                COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IFLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
                       COMMON/SLFIT/ICR12,ICHNG                                        /SLFIT/     2
                       COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG        30AUG       1
                C      J=NP2**15  CMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)  /DISKT/     3
                       CALL ORIGIN (-3.,-4.)                                           CROSS2X    11
     15                CALL ANNOTA(0.,0.,32H PICK CROSSPLOT TYPE,THEN ACCEPT,32,PNAME(6), CROSS2X    12
                      1IGN,6,-0,-0)                                                    CROSS2X    13
                       YOFF=.5                                                         CROSS2X    14
                       XV=1.0                                                          CROSS2X    15
                       YV=0.                                                           CROSS2X    16
     20                DO 10 I=1,4                                                     CROSS2X    17
                       YV=YV-YOFF                                                      CROSS2X    18
                 10    CALL ANNOTA(XV,YV,CTYPE(I),10,PNAME(I),ISIN,I,-0,-0)            CROSS2X    19
                       YV=YV-1.                                                        CROSS2X    20
                       CALL ANNOTA(XV,YV,6HACCEPT,6,PNAME(5),IBUT,5,-0,-0)             CROSS2X    21
     25                CALL GIBUT(0,NCON,IODTR,IODCR)                                  CROSS2X    22
                 11    IF(IODCR.NE.5)GO TO 11                                          CROSS2X    23
                       CALL GIFID (NCON,IODTR,IODCR)                                   CROSS2X    24
                       IF (IODTR.NE.0) GO TO 12                                        CROSS2X    25
                       CALL ERROR#(XV,YV-1.,35HNO SINGLE PICK, FOLLOW INSTRUCTIONS,35, CROSS2X    26
     30                1SHCROSS,6HINFORM)                                              CROSS2X    27
                       GO TO 11                                                        CROSS2X    28
                 12    JT=IODCR                                                        CROSS2X    29
                       CALL ERASE(PNAME,6)                                             CROSS2X    30
                C      ENTER TYPE,INCEP VAR,Y-NAME,                                    CROSS2X    31
     35                YNAME=TYPE=CTYPE(JT)                                            CROSS2X    32
                 1201  CONTINUE                                                        CROSS2X    33
                       NKS=1                                                           FIXMIN      2
                       IF(JT.EQ.3.OR.JT.EQ.5)GO TO 14                                  FIXMIN      3
                       IRP(1)=1HENTER 1 OR                                             CROSS2X    34
     40                IRP(2)=1HH MORE VALU                                            CROSS2X    35
                       IRP(3)=1HES OF THE                                              CROSS2X    36
                       IRP(4)=1HINDEPENDEN                                             CROSS2X    37
                       IRP(5)=1HT VARIABLE                                             CROSS2X    38
                  5    CALL ENTER(IRP,50,0.,0.,AXK,NKS)                                14JUNE     40
     45                IF(NKS.LT.1)GO TO 5                                             CROSS2X    40
                    14 CONTINUE                                                        CROSS2X    41
                       IF (IFLTPTR.EQ.0) GO TO 100                                     CROSS2X    42
                       CALL WINDEX                                                     CROSS2X    43
                       ICR12=3                                                         SLOPES     18
     50                IF(JT.NE.2)GO TO 99                                             SLOPES     19
                       CALL CMGET(ICOM1+11,IFLTPTR,ICV8D)                              SLOPES     20
                       CALL CMGET(ICOM1+11,ICV9D,IFITBD)                               SLOPES     21
                       CALL CMGET(ICOM1+1,IFIT9D,ITYPE)                                SLOPES     22
                       IF(ITYPE.NE.6HAITKEN)GO TO 99                                   SLOPES     23
     55                ICHNG=1                                                         SLOPES     24
                       ICR12=2                                                         SLOPES     25
                       CALL GETSKC(5HDOVER)                                            SLOPES     26
```

```
      99    CONTINUE                                                          SLOPES    27
            CALL AETSKC (6HCROSS2)                                            CROSS2X   44
60          100 CALL FRRORM(XV,YV-1.,17HNULL PLOT POINTER ,17,5HCROSS,6HINFORM)  CROSS2X   45
            CALL AERTRN                                                       LASTBUX    6
            END                                                               CROSS2X   47
```

```
      SUBROUTINE RWPLGR(IBEAD1,INEXT,IPREV,IONE)                        RWPLGR     2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**10  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
5     EXTERNAL SUBGET                                                   RWPLGR     5
C     IONE=1 MEANS ONLY ONE PLOT IN GROUP                               RWPLGR     6
      IONE=1                                                            RWPLGR     7
      CALL DMGET(ICOM1+IPREV,IBEAD1,IPR)                                RWPLGR     8
      IF(IPR.NE.0)GO TO 10                                              RWPLGR     9
10    CALL DMGET(ICOM1+INEXT,IBEAD1,NEXT)                               RWPLGR    10
      IF(NEXT.NE.0)RETURN                                               RWPLGR    11
      IONE=1                                                            RWPLGR    12
      RETURN                                                            RWPLGR    13
   10 CALL SPLICE(IBEAD1,SUBGET,IPREV,0,IPREV,INEXT,0)                  RWPLGR    14
15    END                                                               RWPLGR    15
```

```
1               SUBROUTINE SUBGET(IBEAD1,IBEAD,IVALU)                    RWPLGR     16
                IBEAD1=IBEAD                                             RWPLGR     17
                END                                                      RWPLGR     18
```

```
      SUBROUTINE ENTER(TEXT,NT,XV,YV,RESULTS,NRES)                      ENTER      2
C     DISPLAYS A LINE OF NT CHARACTERS FROM ARRAY TEXT AT LOCATION (XV,Y ENTER      3
C     A LIGHT REGISTER PERMITS INPUTTING A STRING OF UP TO 80 CHARACTERS ENTER      4
C     INPUT IS NUMERIC DATA SEPARATED BY COMMAS                         ENTER      5
C     RETURNED IS NC, THE NUMBER OF CHARACTERS ENTERED                  ENTER      6
C     DECODED NUMERIC VALUES ARE RETURNED THROUGH THE ARRAY RESULTS     ENTER      7
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                /PARMSV/   2
      DIMENSION TEXT(1),RESULTS(1),DELIM(3),BCD(8)                      ENTER      9
      DATA DELIM/1L$,1L ,1L,/,NDL/3/                                    ENTER     10
      CALL ANNOTA(XV,YV,TEXT,NT,IDAD,1,99,-0,-0)                        ENTER     11
      IH=XV*DGUPIN+XORDGU                                               10MAY77   72
      IV=YV*DGUPIN+YORDGU-50.                                           10MAY77   73
      CALL GIEOM (NCON,IEOM,IBUT,200)                                   CY8777    30
  298 NC=40                                                             10MAY77   74
  299 CALL GIANS(NCON,NC,IH,IV)                                         ENTER     15
  300 CALL GIBUT(0,NCON,IDUT,IDDC)                                      ENTER     16
      IF(IDDC.NE.200)GO TO 300                                          ENTER     17
      CALL GIANE(NCON,NC,BCD)                                           ENTER     18
      CALL STRING (1,DELIM,1,BCD,NC+1)                                  CY8777    32
      CALL SELECT(BCD,8,DELIM,NDL,RESULTS,NRES)                         ENTER     19
      CALL ERCK,RETURNS(298)                                            ENTER     20
      CALL GIERAS(IDAD)                                                 ENTER     21
      END                                                               ENTER     22
```

```
      SUBROUTINE SELECT(BCD,NW,DELIM,NDL,RESULTS,NRES)                  SELECT     2
C  FOR INFORMATION CONTACT T. ROWLAND  4950/ADDS  56247                 SELECT     3
C                                                                       SELECT     4
C     DIMENSION BCD(NW),DELIM(NDL),RESULTS(NRES)                        SELECT     5
C                                                                       SELECT     6
C                                                                       SELECT     7
C     SELECT DECODES A STRING OF FREE FIELD INFORMATION WHERE           SELECT     8
C                                                                       SELECT     9
C          BCD IS INPUT ARRAY OF LENGTH NW CONTAINING A STRING          SELECT    10
C        OF FREE FIELD INFORMATION.                                     SELECT    11
C                                                                       SELECT    12
C          DELIM IS INPUT ARRAY OF LENGTH NDL CONTAINING LEFT           SELECT    13
C        JUSTIFIED, ZERO-FILLED DELIMITER CHARACTERS.                   SELECT    14
C                                                                       SELECT    15
C          DELIM(1) MUST CONTAIN THE DELIMITER USED AS AN               SELECT    16
C        END-OF-STRING INDICATOR.                                       SELECT    17
C                                                                       SELECT    18
C          RESULTS IS A RETURNED ARRAY OF LENGTH NRES CONTAINING        SELECT    19
C        DECODED FLOATING-POINT NUMBERS.                                SELECT    20
C                                                                       SELECT    21
C     CALLS SUBROUTINE GETCH(GET I-TH CHARACTER OF J-TH WORD)           SELECT    22
C     AND   SUBROUTINE PUTCH (PUT CHARACTER IN I-TH POSITION OF J-TH WOR SELECT    23
C                                                                       SELECT    24
C                                                                       SELECT    25
C                                                                       SELECT    26
      DIMENSION BCD(1),DELIM(1),RESULTS(1),TEMP(2)                      SELECT    27
      INTEGER C,DELIM                                                   SELECT    28
      TEMP(1)=TEMP(2)=0                                                 SELECT    29
      NC=0                                                              SELECT    30
      II=JJ=K=1                                                         SELECT    31
      DO 90 J=1,NW                                                      SELECT    32
      DO 90 I=1,10                                                      SELECT    33
      CALL GETCH(C,BCD,I,J)                                             SELECT    34
      DO 5 LLL=1,NDL                                                    SELECT    35
      IF(C.EQ.DELIM(LLL))GO TO 20                                       SELECT    36
    5 CONTINUE                                                          SELECT    37
      NC=NC*1                                                           SELECT    38
      CALL PUTCH(C,TEMP,II,JJ)                                          SELECT    39
      II=II+1                                                           SELECT    40
      IF(II.LE.10)GO TO 90                                              SELECT    41
      II=1                                                              SELECT    42
      JJ=2                                                              SELECT    43
      GO TO 90 .                                                        SELECT    44
   20 IF(NC.EQ.0)GO TO 90                                               SELECT    45
      ENCODE(10,7,FMT)NC                                                SELECT    46
    7 FORMAT(2H(E,I2,3H.0))                                             SELECT    47
      DECODE(NC,FMT,TEMP)RESULTS(K)                                     SELECT    48
      K=K+1                                                             SELECT    49
      IF(C.NE.DELIM(1))GO TO 30                                         SELECT    50
   85 NRES=K-1                                                          SELECT    51
      RETURN                                                            SELECT    52
   30 II=1                                                              SELECT    53
      JJ=1                                                              SELECT    54
      TEMP(1)=TEMP(2)=0                                                 SELECT    55
      NC=0                                                              SELECT    56
   90 CONTINUE                                                          SELECT    57
      NRES=K-1                                                          SELECT    58
```

```
      RETURN
      END
```

SELECT 59
SELECT 60

```
      SUBROUTINE GETCH(L,W,I,J)                                          GETCH    2
C                                                                        GETCH    3
C     GETCH RETURNS THE I-TH CHARACTER OF WORD W(J) LEFT JUSTIFIED       GETCH    4
C        IN LOCATION L WITH ZERO FILL.                                   GETCH    5
C                                                                        GETCH    6
      DIMENSION W(J)                                                     GETCH    7
      DATA MSK/77000000000000000000B/                                    GETCH    8
      K=6*(I-1)                                                          GETCH    9
      L=MSK.AND.SHIFT(W(J),K)                                            GETCH   10
      END                                                                GETCH   11
```

```
      SUBROUTINE PUTCH(CH,W,I,J)                                        PUTCH    2
C                                                                       PUTCH    3
C     PUTS CHARACTER CH (LEFT JUSTIFIED) IN POSITION I OF WORD(J)       PUTCH    4
C                                                                       PUTCH    5
C     CAUTION, I-TH POSITION OF J-TH WORD MUST CONTAIN BINARY ZEROS.    PUTCH    6
C                                                                       PUTCH    7
      DIMENSION W(J)                                                    PUTCH    8
      DATA MSK/77000000000000000000B/                                   PUTCH    9
      K=6*(11-I)                                                        PUTCH   10
      W(J)=W(J).OR.SHIFT(CH.AND.MSK,K)                                  PUTCH   11
      END                                                               PUTCH   12
```

```
      OVERLAY (4,1)                                                    INHALE     23
      PROGRAM NXTPLOT                                                  INHALE     24
C                                                                      NXTPLOT     3
C         THIS ROUTINE ERASES THE CURRENT PLOT AND DISPLAYS THE NEXT   NXTPLOT     4
C         "CROSS PLOT" PLOT OR "TRIM" PLOT (WHICHEVER IS APPROPRIATE). NXTPLOT     5
C              IPLTPTR - I/O - AT THE TIME CONTROL PASSES TO NXTPLOT    NXTPLOT     6
C                        IPLTPTR IS THE POINTER TO THE PLOT BEAD OF THE NXTPLOT     7
C                        PLOT TO BE ERASED. WHEN CONTROL RETURNS TO THE NXTPLOT     8
C                        CALLING PROGRAM IPLTPTR IS THE POINTER TO THE  NXTPLOT     9
C                        PLOT BEAD OF THE NEW PLOT, IF ONE EXISTS.      NXTPLOT    10
C                        IF NOT, IPLTPTR IS UNCHANGED.                  NXTPLOT    11
C                                                                       NXTPLOT    12
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IOSPEC(19),IOCF(41), /DISPLAY/     2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IOMENU(20),         6JULY77     2
     X               IOINSTR(19),ICDATA8(20,4,2),NCIP,IOBLOCK(20),NCIPN 1SEPT      1
      COMMON/OPTION/IOPTION,ISHINE                                     LASTBUX     1
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,     /PLTBEAD/    2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,      /PLTBEAD/    3
     X               IPFMN,IPFMV                                       /PLTBEAD/    4
      COMMON /SAVCPPT/ NEXCPPT                                         /SAVCPPT/    2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/     2
      COMMON /NXS/ NEXTSET,IERR                                        NEWFACE    27
      LOGICAL NEXTSET                                                  INHALE     27
      IERR=0                                                           NEWFACE    28
      IF (NEXTSET) GO TO 5                                             INHALE     28
C                                                                      NXTPLOT    16
C         GET THE POINTER TO THE NEXT PLOT BEAD.                       NXTPLOT    17
C                                                                      NXTPLOT    18
      IF(IOPTION .EQ. 10HCROSS PLOT) CALL DMGET(IPNEXTC,IPLTPTR,IBEAD) NXTPLOT    19
      IF(IOPTION .EQ. 4HTRIM) CALL DMGET(IPNEXTT,IPLTPTR,IBEAD)        NXTPLOT    20
C                                                                      NXTPLOT    21
C         CHECK TO SEE IF THERE IS ANOTHER PLOT. IF NOT,               NXTPLOT    22
C          DISPLAY AN INFORMATIVE ERROR MESSAGE.                       NXTPLOT    23
C                                                                      NXTPLOT    24
      IF(IBEAD .NE. 0) GO TO 10                                        NXTPLOT    25
      CALL ERRORM(-4.,8.,38HTHERE ARE NO MORE PLOTS IN THIS GROUP.,38, NXTPLOT    26
     X            7HNXTPLOT,6HINFORM)                                  NXTPLOT    27
      IERR=1                                                           NEWFACE    29
      CALL AERTRN                                                      INHALE     29
C                                                                      14JUNE     44
C         ENTRY POINT FOR MOVING TO NEXT SET (TRIM ONLY)               14JUNE     45
C                                                                      14JUNE     46
    5 IBEAD=NEXCPPT                                                    INHALE     30
      NEXTSET=.FALSE.                                                  INHALE     31
      IF (IBEAD.NE.0) GO TO 10                                         14JUNE     51
      CALL ERRORM (-4.,8.,37HTHERE ARE NO MORE SETS IN THIS GROUP.,    14JUNE     52
     1     37,7HNXTPLOT,6HINFORM)                                      14JUNE     53
      IERR=1                                                           NEWFACE    30
      CALL AERTRN                                                      INHALE     32
C                                                                      NXTPLOT    29
C         THERE IS ANOTHER PLOT. ERASE THE CURVES, THE                 25MAY76     7
C          CURVE FITS, AND THE CURVE DESCRIPTIONS.                     25MAY76     8
C                                                                      NXTPLOT    32
   10 IPLTPTR = IBEAD                                                  NXTPLOT    33
      IF (IBEAD.EQ.NEXCPPT) CALL DMGET (IPNEXTC,IBEAD,NEXCPPT)         14JUNE     56
      NLOC=NCIP*4                                                      1SEPT      7
      CALL TERASE (IDDAD,NLOC)                                         ADPTRRE    14
```

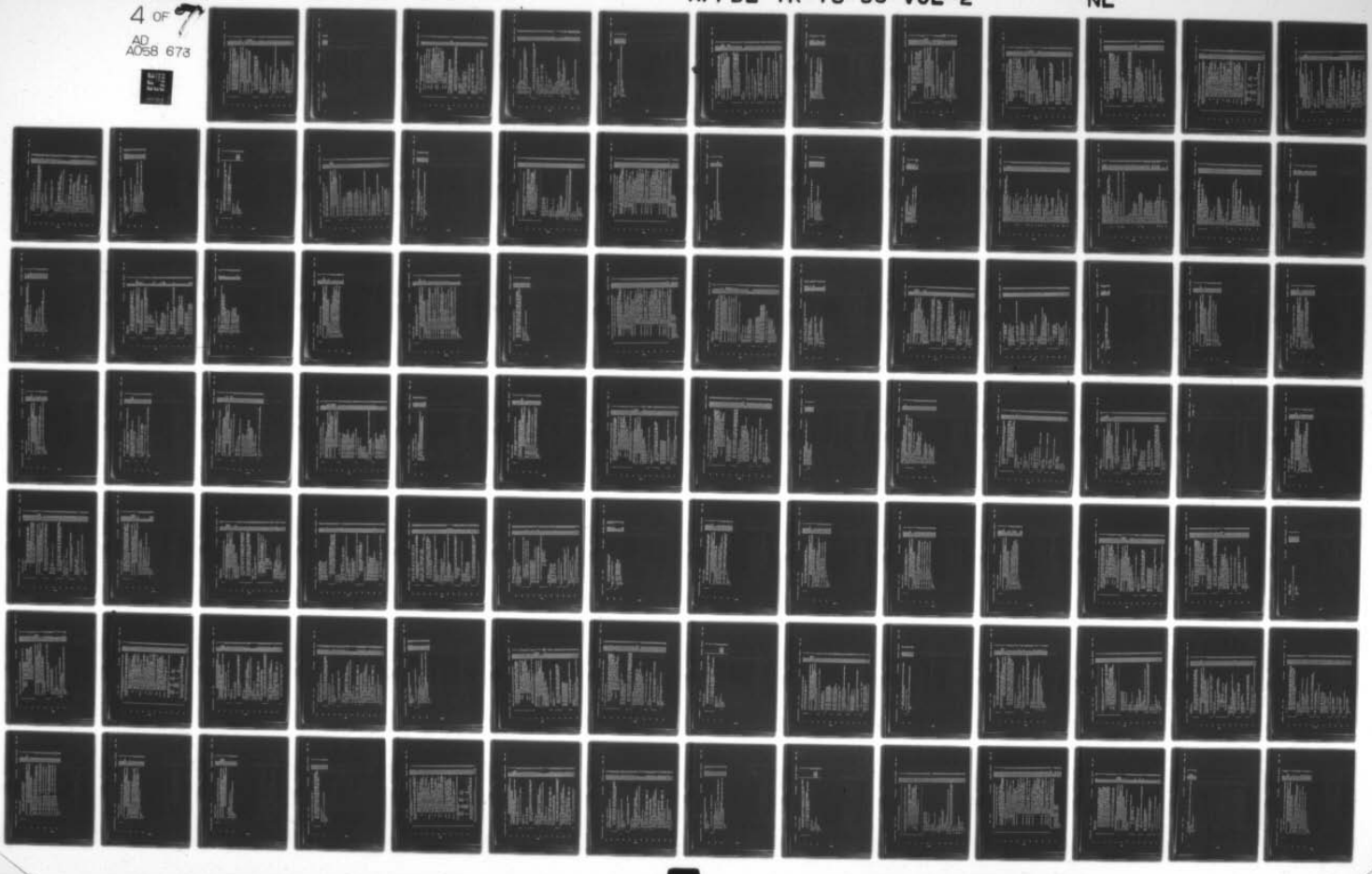
```
                      NN = 0                                                    NXTPLOT    35
                      CALL TERASE (IOCF,NLOC)                                   NSROC       6
60                    IOCF(41)=0                                                25MAY76    10
                      CALL TERASE(IOCD,NCIP)                                    NXTPLOT    36
                      CALL TERASE (IOTITLE,4)                                   14JUNE     57
                C                                                               18JUNE76    1
                C         INITIALIZE CURVE AND FIT HEAD POINTERS                18JUNE76    2
65              C         OF THE OLD PLOTS                                      18JUNE76    3
                C                                                               18JUNE76    4
                      DO 15 J=1,NCIP                                            18JUNE76    5
                      IODATA8(J,1,2)=8HINACTIVE                                 18JUNE76    6
                      IODATA8(J,4,1)=0                                          18JUNE76    7
70                    IODATA8(J,4,2)=0                                          18JUNE76    8
                   15 CONTINUE                                                  16JUNE76    9
                C     IF (IOPTION.EQ.4HTRIM) GO TO 20                           3FEB        9
                C                                                               NXTPLOT    38
                C         THE OPTION IS CROSS PLOT. IF MULTIPLIERS ARE DISPLAYED, NXTPLOT    39
75              C         ERASE THEM AND DISPLAY THE MULTIPLIERS FOR THE NEXT PLOT. NXTPLOT   40
                C                                                               NXTPLOT    41
                      IF(IDSPEC(19).EQ.0)GO TO 20                               5OCT       29
                      NMS=0                                                     2NOV       43
                      CALL TERASE(IOSPEC(IDSPEC(18) + 1),IDSPEC(19) - IDSPEC(18)) NXTPLOT  43
80                    CALL ORIGIN(-5.,8.5)                                      5MAY76     11
                      CALL DISMULT(IPLTPTR)                                     5MAY76     12
                      CALL ORIGIN(-5.,-3.)                                      5MAY76     13
                      GO TO 20                                                  5OCT       30
                C                                                               NXTPLOT    46
85              C         THE OPTION IS TRIM. ERASE THE AXES AND                NXTPLOT    47
                C         LABELS AND DISPLAY A NEW AXIS SYSTEM.                 NXTPLOT    48
                C                                                               NXTPLOT    49
                   20 CALL TERASE(IOAXIS,4)                                     NXTPLOT    50
                      CALL ORIGIN (-5.,-3.)                                     1SEPT       8
90                    CALL DISAXES(IPLTPTR)                                     NXTPLOT    51
                      CALL LBLAXES(IPLTPTR)                                     NXTPLOT    52
                      CALL DISTITL (IPLTPTR)                                    14JUNE     58
                C                                                               NXTPLOT    53
                C         DISPLAY THE NEXT PLOT.                                NXTPLOT    54
95              C                                                               NXTPLOT    55
                   30 CALL FRTRN                                                INHALE     33
                      END                                                       NXTPLOT    58
```

```
      SUBROUTINE DISAXES(IPLTPTR)                                        DISAXES     2
C                                                                        DISAXES     3
C     THIS ROUTINE:                                                      DISAXES     4
C          1 - RETRIEVES AND COMPUTES ALL PLOT SCALING INFORMATION.      DISAXES     5
C          2 - DISPLAYS A NINE INCH X-AXIS, A SIX INCH Y-AXIS, AND       DISAXES     6
C              SCALE INFORMATION FOR BOTH.                               DISAXES     7
C                                                                        DISAXES     8
C     PRIMARY INPUT TO THE ROUTINE IS A POINTER TO THE PLOT BEAD OF      DISAXES     9
C     THE PLOT FOR WHICH AXES ARE TO BE DRAWN. THE AXES SCALE DATA       DISAXES    10
C     IS OUTPUT IN COMMON/AXSCALE/.                                      DISAXES    11
C                                                                        DISAXES    12
      COMMON/AXSBEAD/IAXSXM,IAXSXI,IAXSY1M,IAXSY1I,IAXSY2M,IAXSY2I,      /AXSBEAD/    2
     X               IAXSY3M,IAXSY3I                                     /AXSBEAD/    3
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/    2
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/    2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77      2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT        1
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/    2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/    3
     X               IPFMN,IPFMV                                        /PLTBEAD/    4
C                                                                        DISAXES    17
C     GET THE MINIMUMS AND INCREMENTS FROM THE AXIS                      DISAXES    18
C     SCALE BEAD AND COMPUTE THE MAXIMUMS.                               DISAXES    19
C                                                                        DISAXES    20
      CALL DMGET(IPAXISS,IPLTPTR,IAXSPTR)                                DISAXES    21
      CALL DMGET(IAXSXM,IAXSPTR,XMIN)                                    DISAXES    22
      CALL DMGET(IAXSXI,IAXSPTR,XINCR)                                   DISAXES    23
      CALL DMGET(IAXSY1M,IAXSPTR,YMIN)                                   DISAXES    24
      CALL DMGET(IAXSY1I,IAXSPTR,YINCR)                                  DISAXES    25
      XMAX = XMIN + 9. * XINCR                                           DISAXES    26
      YMAX = YMIN + 6. * YINCR                                           DISAXES    27
C                                                                        DISAXES    28
C     IF INCR IS NEGATIVE, THEN MIN IS IN FACT THE MAXIMUM AND MAX       DISAXES    29
C     IS IN FACT THE MINIMUM. THE FOLLOWING CODE INSURES THAT MIN        DISAXES    30
C     AND MAX ARE IN FACT THE MINIMUM AND MAXIMUM RESPECTIVELY.          DISAXES    31
C                                                                        DISAXES    32
      IF(XMIN .LE. XMAX) GO TO 10                                        DISAXES    33
      TEMP=XMIN                                                          DISAXES    34
      XMIN=XMAX                                                          DISAXES    35
      XMAX=TEMP                                                          DISAXES    36
C                                                                        DISAXES    37
   10 IF(YMIN .LE. YMAX) GO TO 20                                        DISAXES    38
      TEMP = YMIN                                                        DISAXES    39
      YMIN = YMAX                                                        DISAXES    40
      YMAX = TEMP                                                        DISAXES    41
C                                                                        DISAXES    42
   20 XINCR=ABS(XINCR)                                                   DISAXES    43
      YINCR = ABS(YINCR)                                                 DISAXES    44
C                                                                        DISAXES    45
C     DISPLAY THE AXES.                                                  DISAXES    46
C                                                                        DISAXES    47
      CALL DMSET (IAXSXM,IAXSPTR,XMIN)                                   20SEPT       1
      CALL DMSET (IAXSXI,IAXSPTR,XINCR)                                  20SEPT       2
      CALL DMSET (IAXSY1M,IAXSPTR,YMIN)                                  20SEPT       3
      CALL DMSET (IAXSY1I,IAXSPTR,YINCR)                                 20SEPT       4
      CALL GAXIS(0.,1.,9.,0.,XMIN,XINCR,-1,IDAXIS(1),-0,-0)              DISAXES    48
      CALL GAXIS(0.,1.,6.,90.,YMIN,YINCR,1,IDAXIS(2),-0,-0)              DISAXES    49
```

```
      RETURN                                                    DISAXES    50
      END                                                       DISAXES    51
```

AD-A058 673 DAYTON UNIV OHIO RESEARCH INST F/G 9/2
PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). VOLUME II--ETC(U)
MAY 78 J G JENSEN F33615-76-C-3076
UNCLASSIFIED UDR-TR-78-34 AFFDL-TR-78-55-VOL-2 NL

4 OF 7
AD A058 673



```
      SUBROUTINE DISMULT(IPLTPTR)                                        5MAY76     1
C                                                                        DISMULT    3
C         THIS ROUTINE DISPLAYS THE MULTIPLIERS AT THE TOP OF THE PLOT.  DISMULT    4
C                                                                        DISMULT    5
C         WORD IDSPEC(19) IS A STATUS WORD FOR THE MULTIPLIERS. IF IT IS DISMULT    6
C         ZERO, THEN NO MULTIPLIERS ARE DISPLAYED. IF IT IS NOT ZERO,    DISMULT    7
C         THEN MULTIPLIERS ARE DISPLAYED AND IDSPEC(19) CAN BE USED IN   DISMULT    8
C         CONJUNCTION WITH IDSPEC(18) TO ERASE THEM. THE DISPLAY         DISMULT    9
C         ADDRESSES OF THE MULTIPLIERS ARE STORED IN                     DISMULT   10
C         IDSPEC(IDSPEC(18) + 1) THROUGH IDSPEC(IDSPEC(19)).             DISMULT   11
C                                                                        DISMULT   12
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CY8777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77    3
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,     /PGBEAD/    2
     X              IPGLCA                                              /PGBEAD/    3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/   2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,        /PLTBEAD/   3
     X               IPFMN,IPFMV                                         /PLTBEAD/   4
      INTEGER RECORD(2)                                                  DISMULT   17
C                                                                        DISMULT   18
C         GET THE NUMBER OF MULTIPLIERS FROM THE PLOT BEAD.              DISMULT   19
C         THE NUMBER OF LINES OF CONDITION SPECIFICATIONS                5MAY76     2
C         WHICH HAVE BEEN DISPLAYED IS IN IDSPEC(18). CHECK              5MAY76     3
C         TO SEE IF THE MULTIPLIERS CAN BE DISPLAYED.                    5MAY76     4
C                                                                        DISMULT   20
      CALL DMGET(IPAMULT,IPLTPTR,NMULT)                                  DISMULT   21
      IF(NMULT .GT. 0) GO TO 10                                          DISMULT   22
      IDSPEC(19) = 0                                                     DISMULT   23
      RETURN                                                             DISMULT   24
   10 LINECNT = IDSPEC(18)                                               5MAY76     5
      IF((LINECNT + NMULT +1) .LE. 16) GO TO 20                          5MAY76     6
      IDSPEC(19) = 0                                                     DISMULT   30
      RETURN                                                             DISMULT   31
C                                                                        DISMULT   32
C         DISPLAY THE MULTIPLIER(S) HEADING.                             DISMULT   33
C                                                                        DISMULT   34
   20 LINECNT = LINECNT + 1                                              DISMULT   35
      Y = (-.25) * FLOAT(LINECNT)                                        DISMULT   36
      CALL ANNOTA(.2,Y,13HMULTIPLIER(S),13,IDSPEC(LINECNT),IGN,-0,-0,-0) DISMULT   37
C                                                                        DISMULT   38
C         LOOP THROUGH THE MULTIPLIERS.                                  DISMULT   39
C                                                                        DISMULT   40
      DO 30 I=1,NMULT                                                    DISMULT   41
C                                                                        DISMULT   42
C         GET THE MULTIPLIER NAME AND VALUE FROM THE                     DISMULT   43
C         PLOT BEAD, ENCODE IT, AND DISPLAY IT.                          DISMULT   44
C                                                                        DISMULT   45
      CALL DMGET(IPFMN+2*I-2,IPLTPTR,NAME)                               DISMULT   46
      CALL DMGET(IPFMV+2*I-2,IPLTPTR,VALUE)                              DISMULT   47
      ENCODE(20,1000,RECORD) NAME,VALUE                                  DISMULT   48
 1000 FORMAT(A7,"=",G12.6)                                               DISMULT   49
      LINECNT = LINECNT + 1                                              DISMULT   50
      Y = (-.25) * FLOAT(LINECNT)                                        DISMULT   51
      CALL ANNOTA(.2,Y,RECORD,20,IDSPEC(LINECNT),IGN,-0,-0,-0)           DISMULT   52
```

```
      30 CONTINUE                                                    DISMULT    53
         IOSPEC(19) = LINECNT                                        DISMULT    54
      60 RETURN                                                      DISMULT    55
         END                                                         DISMULT    56
```

```
      SUBROUTINE GAXIS(XINCH,YINCH,XLENIN,THETA,XMIN,DELTA,                GAXIS      2
     X                 ISIGN,IDDAD,IDDT,IDDC)                              GAXIS      3
C                                                                          GAXIS      4
C     GAXIS IS A GRAPHICS ROUTINE TO DRAW A LINEAR AXIS.                   GAXIS      5
C                                                                          GAXIS      6
C       XINCH - THE X-COORDINATE IN INCHES OF THE STARTING POINT           GAXIS      7
C               OF THE AXIS.                                               GAXIS      8
C       YINCH - THE Y-COORDINATE IN INCHES OF THE STARTING POINT           GAXIS      9
C               OF THE AXIS.                                               GAXIS     10
C      XLENIN - THE LENGTH OF THE AXIS IN INCHES.                          GAXIS     11
C       THETA - THE ANGLE OF THE AXIS MEASURED IN DEGREES COUNTER -        GAXIS     12
C               CLOCKWISE FROM THE NORMAL POSITIVE X-AXIS. Y CAN           GAXIS     13
C               EQUAL 0., 90., 180., OR 270. DEGREES ONLY.                 GAXIS     14
C        XMIN - THE COORDINATE VALUE AT THE FIRST POINT OF THE AXIS.       GAXIS     15
C       DELTA - COORDINATE UNITS PER INCH ALONG THE AXIS.                  GAXIS     16
C       ISIGN - -1 IF THE COORDINATE UNITS ARE TO BE DISPLAYED ON          GAXIS     17
C                  THE CLOCKWISE SIDE OF THE AXIS.                         GAXIS     18
C               +1 IF THE COORDINATE UNITS ARE TO BE DISPLAYED ON          GAXIS     19
C                  THE COUNTER - CLOCKWISE SIDE OF THE AXIS.               GAXIS     20
C       IDDAD - THE SYSTEM DEFINED ASSOCIATIVE ADDRESS OF THE              GAXIS     21
C               DISPLAY.                                                   GAXIS     22
C        IDDT - ID DISPLAY TYPE.                                           GAXIS     23
C        IDDC - ID DISPLAY CODE.                                           GAXIS     24
C                                                                          GAXIS     25
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                     CV0777     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                     6JUL77     3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                   /PARMSV/   2
      DATA IFMT1,IFMT2/6H(G9.3),6H(F9.4)/                                  3FE8       2
C                                                                          GAXIS     28
C     USE THETA AND DGUPIN TO CALCULATE THE HORIZONTAL AND VERTICAL        GAXIS     29
C     COMPONENTS OF THE UNIT DISPLACEMENT VECTOR. THE RESULTS ARE IN       GAXIS     30
C     DIGIGRAPHIC UNITS PER INCH.                                          GAXIS     31
C                                                                          GAXIS     32
      THR = .01745329 * THETA                                              GAXIS     33
      STH = SIN(THR)                                                       GAXIS     34
      CTH = COS(THR)                                                       GAXIS     35
      IF(ABS(STH) .LE. .0001) STH = 0.                                     GAXIS     36
      IF(ABS(CTH) .LE. .0001) CTH = 0.                                     GAXIS     37
      IHDEL = IFIX(CTH * DGUPIN + .5)                                      GAXIS     38
      IVDEL = IFIX(STH * DGUPIN + .5)                                      GAXIS     39
C                                                                          GAXIS     40
C     CALCULATE THE COORDINATES OF THE ORIGIN                              GAXIS     41
C     IN DGUS AND DISPLAY A PLUS THERE.                                    GAXIS     42
C                                                                          GAXIS     43
      IXDGU = IFIX(XINCH * DGUPIN + XORDGU + .5)                           GAXIS     44
      IYDGU = IFIX(YINCH * DGUPIN + YORDGU + .5)                           GAXIS     45
      NBYTE = 0                                                            GAXIS     46
      ICODE = 0028                                                         GAXIS     47
      CALL GURSET(IXDGU - 8 ,IYDGU - 12,ICODE,IBUF,NBYTE,MBYTE)            GAXIS     48
      CALL GUAN(1H+,1,IBUF,NBYTE,MBYTE)                                    GAXIS     49
C                                                                          GAXIS     50
C     DRAW THE AXIS AN INCH AT A TIME. FOLLOW EACH ONE INCH LINE           GAXIS     51
C     SEGMENT WITH A PLUS AT EACH NODE. THE CODE TRUNCATES THE             GAXIS     52
C     LENGTH OF THE AXIS TO AN INTEGER.                                    GAXIS     53
C                                                                          GAXIS     54
      NTP = IFIX(XLENIN + .0001)                                           GAXIS     55
      IH1 = IXDGU                                                          GAXIS     56
```

```
                IV1 = IYOGU                                                     GAXIS      57
                DO 10 I=1, NIN                                                  GAXIS      58
 60             CALL GURSET(IH1,IV1,ICODE,IBUF,NBYTE,MBYTE)                     GAXIS      59
                IH2 = IH1 + IHDEL                                               GAXIS      60
                IV2 = IV1 + IVDEL                                               GAXIS      61
                CALL GUSEGS(IH1,IV1,IH2,IV2,IBEAM,ISTYLE,IBUF,NBYTE,MBYTE)      GAXIS      62
                CALL GURSET(IH2 - 5 ,IV2 - 12,ICODE,IBUF,NBYTE,MBYTE)           GAXIS      63
 65             CALL GUAN(1H+,1,IBUF,NBYTE,MBYTE)                               GAXIS      64
                IH1 = IH2                                                       GAXIS      65
                IV1 = IV2                                                       GAXIS      66
             10 CONTINUE                                                        GAXIS      67
          C                                                                     GAXIS      68
 70       C         CALCULATE THE COORDINATES IN DGUS OF THE POINT ON THE SCREEN GAXIS      69
          C         WHERE THE INITIAL COORDINATE VALUE WILL BE DISPLAYED.       GAXIS      70
          C                                                                     GAXIS      71
                IDIR = IFIX(THETA/90. + 1.000001)                               GAXIS      72
                GO TO(20,30,20,30),IDIR                                         GAXIS      73
 75          20 IYADJ = -100                                                    GAXIS      74
                IF(ISIGN .EQ. -1) IYADJ = 100 * ISIGN                           GAXIS      75
                IF(ISIGN .EQ. +1) IYADJ =  50 * ISIGN                           GAXIS      76
                GO TO 40                                                        GAXIS      77
             30 IXADJ = - 250                                                   GAXIS      78
 80             IF(ISIGN .EQ. -1) IXADJ = 50                                    GAXIS      79
                IYADJ = - 12                                                    GAXIS      80
             40 IH1 = IXOGU + IXADJ                                             GAXIS      81
                IV1 = IYOGU + IYADJ                                             GAXIS      82
             80 NNODES = NIN + 1                                                GAXIS     113
 85             VAL=XMIN                                                        SCAXES      1
                IFMT=IFMT2                                                      3FEB        3
                ABMIN=ABS(XMIN)                                                 3FEB        4
                ABDEL=ABS(DELTA)                                                3FEB        5
                IF (ABMIN.LT..0001.OR.ABMIN.GT.100.) IFMT=IFMT1                 3FEB        6
 90             IF (ABDEL.GT.100.) IFMT=IFMT1                                   3FEB        7
                DO 110 I=1,NNODES                                               GAXIS      115
             90 ENCODE (9,IFMT,VALOC) VAL                                       3FEB        8
            100 CALL GURSET(IH1,IV1,ICODE,IBUF,NBYTE,MBYTE)                     GAXIS      124
                CALL GUAN(VALOC,9,IBUF,NBYTE,MBYTE)                             GAXIS      125
 95             VAL=VAL+DELTA                                                   SCAXES      2
                IH1 = IH1 + IHDEL                                               GAXIS      128
                IV1 = IV1 + IVDEL                                               GAXIS      129
            110 CONTINUE                                                        GAXIS      130
          C                                                                     GAXIS      131
100       C         DISPLAY THE AXIS.                                           GAXIS      132
          C                                                                     GAXIS      133
                CALL GIDISP(NCON,IBUF,NBYTE,IODAD,IODT,IODC)                    GAXIS      134
                ICODE = 1028                                                    GAXIS      135
                RETURN                                                          GAXIS      136
105             END                                                             GAXIS      137
```

```
1                OVERLAY (4,2)                                                    DISPLTX     2
                 PROGRAM DISPLTX                                                  DISPLTX     3
           C                                                                      DISPLTX     4
           C        DUMMY TO CALL DISPLOT AND OTHER PLOT ROUTINES                 DISPLTX     5
5          C                                                                      DISPLTX     6
                 COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
                 CALL DISPLOT (IPLTPTR)                                           DISPLTX     8
                 CALL AERTRN                                                      DISPLTX     9
                 END                                                              DISPLTX    10
```

```
      SUBROUTINE DISPLOT(IPLTPTR)                                        DISPLOT    2
C                                                                        DISPLOT    3
C         THIS ROUTINE DISPLAYS ALL OF THE ACTIVE CURVES IN A PLOT.      DISPLOT    4
C         THE PRIMARY INPUT TO THE ROUTINE IS THE PLOT BEAD POINTER.     DISPLOT    5
C                                                                        DISPLOT    6
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREG,   /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                  /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(99),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(28),           6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIFP  1SEPT     1
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/  2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,        /PLTBEAD/  3
     X               IPFMN,IPFMV                                         /PLTBEAD/  4
      CALL ORIGIN(-5.,-3.)                                               DISPLOT   10
C                                                                        DISPLOT   11
C         INITIALIZE NCIP. NCIP IS THE NUMBER OF CURVES IN A PLOT. MORE  DISPLOT   12
C         SPECIFICALLY, NCIP IS THE NUMBER OF CURVES WHOSE ACCESS DATA   DISPLOT   13
C         IS STORED IN THE CURVE DISPLAY ACCESS TABLE(SEE SUBROUTINE     DISPLOT   14
C         LOCATAB FOR AN EXPLANATION OF THE CURVE DISPLAY ACCESS TABLE).  DISPLOT   15
C         THE MAXIMUM VALUE OF NCIP IS TWENTY, I.E., THERE CAN BE NO     DISPLOT   16
C         MORE THAN TWENTY CURVES PER PLOT.                              DISPLOT   17
C                                                                        DISPLOT   18
      NCIP=0                                                             DISPLOT   19
      DO 5 J=1,41                                                        2NOV      1
    5 IDCF(J)=0                                                          2NOV      2
C                                                                        DISPLOT   20
C         GET THE POINTER TO THE FIRST CURVE BEAD FROM THE PLOT BEAD.    DISPLOT   21
C                                                                        DISPLOT   22
      CALL DMGET(IPCURVE,IPLTPTR,ICRVPTR)                                DISPLOT   23
      GO TO 20                                                           DISPLOT   24
C                                                                        DISPLOT   25
C         GET THE POINTER TO THE NEXT CURVE BEAD.                        DISPLOT   26
C                                                                        DISPLOT   27
   10 CALL DMGET(ICNEXT,ICRVPTR,ICRVPTR)                                 DISPLOT   28
      IF(ICRVPTR .EQ. 0) RETURN                                          DISPLOT   29
C                                                                        DISPLOT   30
C         A CURVE BEAD EXISTS. IS THE CURVE ACTIVE OR INACTIVE?          DISPLOT   31
C             ACTIVE - ISTAT = 0                                         DISPLOT   32
C            INACTIVE - ISTAT = 8HINACTIVE                               14APR76    2
C         A CURVE IS INACTIVE IF AND ONLY IF IT HAS BEEN DELETED DURING  DISPLOT   34
C         AN EDIT.                                                       DISPLOT   35
C                                                                        DISPLOT   36
   20 CALL DMGET(ICSTAT,ICRVPTR,ISTAT)                                   DISPLOT   37
      IF(ISTAT .EQ. 8HINACTIVE) GO TO 10                                 14APR76    3
C                                                                        DISPLOT   39
C         THE CURVE IS ACTIVE. HAS THE CURVE LIMIT BEEN REACHED?         DISPLOT   40
C                                                                        DISPLOT   41
      IF(NCIP .LT. 20) GO TO 30                                          DISPLOT   42
C                                                                        DISPLOT   43
C         THE CURVE LIMIT WAS REACHED ON THE PREVIOUS CURVE.             DISPLOT   44
C                                                                        DISPLOT   45
      CALL ERRORM(0.,7.,72HTHERE ARE MORE THAN TWENTY CURVES IN THE PLOT DISPLOT   46
     X. TWENTY WILL BE DISPLAYED.,72,7HDISPLOT,6HINFORM)                 DISPLOT   47
      RETURN                                                             DISPLOT   48
C                                                                        DISPLOT   49
C         THE CURVE LIMIT HAS NOT BEEN REACHED. DISPLAY THE CURVE AND    DISPLOT   50
C         ITS DESCRIPTION AND UPDATE THE CURVE DISPLAY ACCESS TABLE.     DISPLOT   51
```

```
      C                                                                 DISPLOT   52
         30 CALL DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NCIP+1)        ADPTPRE    6
60       IF(NUMDISP .EQ. 0) GO TO 10                                    DISPLOT   54
         NCIP=NCIP+1                                                    DISPLOT   55
         NCIPP=NCIP                                                     1SEPT     2
         CALL DCRVOFS(ICRVPTR,ISYMBOL,NCIP)                             DISPLOT   56
         CALL UCDATAB(NCIP,1,6HACTIVE,INITSS,NUMDISP,ICRVPTR)           14APR76   4
65       CALL CMGET (ICFIT,ICRVPTR,IFT)                                 13SEPT    2
         IF (IFT.EQ.0) CALL UCDATAB (NCIP,2,8HINACTIVE,0,0,0)           13SEPT    3
         IF (IFT.GT.0) CALL REPLCF (NCIP,ICRVPTR,IFT,0)                 20SEPT    5
         GO TO 10                                                       DISPLOT   58
         END                                                            DISPLOT   59
```

```
      SUBROUTINE DCRVDES(ICRVPTR,ISYMBOL,NCURVE)                          DCRVDES    2
C                                                                         DCRVDES    3
C     THIS ROUTINE BUILDS A CURVE DESCRIPTION AND DISPLAYS IT.            DCRVDES    4
C          ICRVPTR - THE CURVE BEAD POINTER.                              DCRVDES    5
C          ISYMBOL - THE CURVE PLOT SYMBOL.                               DCRVDES    6
C           NCURVE - THE NUMBER OF THE CURVE. THE CURVES ARE              DCRVDES    7
C                    NUMBERED CHRONOLOGICALLY AS THEY APPEAR              DCRVDES    8
C                    ON THE SCREEN. THAT NUMBER IS USED IN                DCRVDES    9
C                    THIS SUBROUTINE TO DETERMINE THE POSITION            DCRVDES   10
C                    ON THE SCREEN WHERE THE CURVE DESCRIPTION            DCRVDES   11
C                    WILL BE DISPLAYED.                                   DCRVDES   12
C                                                                         DCRVDES   13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,    /CRVBEAD/   2
     X              ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                    /CRVBEAD/   3
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),   /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),           6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT      1
      COMMON /GP/ IBUF(128),NCOM,NBYTE,MBYTE,IGN,ISIN,                    CYBTTT     1
     Y     ISTR,IBUT,IMARK,ICODE,IBEAM,ICTYLE,IEOM,CPTO                   6JULY77    3
      INTEGER RECORD(3)                                                   DCRVDES   17
C                                                                         DCRVDES   18
C     GET THE PARAMETER NAME AND VALUE FROM THE CURVE BEAD, ENCODE        DCRVDES   19
C     THE CURVE DESCRIPTION, AND DISPLAY IT.                              DCRVDES   20
C                                                                         DCRVDES   21
      CALL DMGET(ICPARAN,ICRVPTR,NAME)                                    DCRVDES   22
      CALL DMGET(ICPARAV,ICRVPTR,VALUE)                                   DCRVDES   23
      IDEC=8                                                              FIXINH
      DO 10 J=1,9                                                         FIXINH
      IF (VALUE/10.**(J-1).LE.1.) GO TO 20                                FIXINH    4
   10 IDEC=IDEC-1                                                         FIXINH    5
      IDEC=0                                                              FIXINH    6
   20 ENCODE (25,1000,RECORD) ISYMBOL,NAME,IDEC,VALUE                     FIXINH    7
 1000 FORMAT(I1,1X,A10,1X,F12.=)                                          FIXINH    8
      Y = 8.0 - .5 * FLOAT(NCURVE)                                        DCRVDES   26
      CALL ANNOTA (10.43,Y,RECORD,25,IDCD(NCURVE),IGN,NCURVE,-0,-0)       6JULY77    6
      RETURN                                                              DCRVDES   28
      END                                                                 DCRVDES   29
```

```
      SUBROUTINE DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NPLOT)                ADPTPRE    1
C                                                                              DISCRVE    3
C     THIS ROUTINE DISPLAYS THE CURVE(SET OF POINTS) INDICATED BY              DISCRVE    4
C     "ICRVPTR". THE PRIMARY INPUTS TO THE ROUTINE ARE THE POINTER             DISCRVE    5
C     TO THE CURVE BEAD AND THE PLOT SCALE DATA. PRIMARY OUTPUTS               DISCRVE    6
C     FROM THE ROUTINE ARE:                                                    DISCRVE    7
C          ISYMBOL - THE SYMBOL USED TO DISTINGUISH THIS CURVE FROM            DISCRVE    8
C                    OTHER CURVES ON THE PLOT.                                 DISCRVE    9
C           INITSS - THE DISPLAY ADDRESSES FOR THE CURVE ARE STORED            DISCRVE   10
C                    IN ARRAY IODAD. IODAD(INITSS) CONTAINS THE                DISCRVE   11
C                    DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED             DISCRVE   12
C                    TO EXHIBIT THE CURVE.(INITSS-INITIAL SUBSCRIPT)           DISCRVE   13
C          NUMDISP - NUMDISP CONTAINS THE NUMBER OF DISPLAYS                   DISCRVE   14
C                    REQUIRED TO EXHIBIT THE CURVE.                            DISCRVE   15
C          NPLOT   - NPLOT CONTAINS THE NUMBER OF THE                          ADPTPRE    2
C                    PLOT IN THE TABLE                                         ADPTPRE    3
C     INITSS AND NUMDISP ARE NEEDED TO ERASE THE CURVE. THEY WILL BE           DISCRVE   16
C     STORED IN THE CURVE DISPLAY ACCESS TABLE.                                DISCRVE   17
C                                                                              DISCRVE   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                           /AXSCALE/  2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,         /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                        /CRVBEAD/  3
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),        /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IDMENU(20),                 6JULY77    2
     X               IOINSTF(19),IODATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM        1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                         CYB777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                        6JULY77    3
      COMMON/PARMSV/XORUGU,YORUGU,OSUPIN                                       /PARMSV/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                    /POINTS/   2
      COMMON /SYMTAB/ ISYMTAB(20)                                              /SYMTAB/   2
      DIMENSION MESG(7)                                                        9JULY76    1
C                                                                              DISCRVE   28
C     RECORD THE INITIAL VALUE OF THE IODAD SUBSCRIPT.                         DISCRVE   29
C                                                                              DISCRVE   30
      INITSS=(NPLOT-1)*4+1                                                     ADPTPRE    4
      NN=INITSS-1                                                              ADPTPRE    5
C                                                                              DISCRVE   32
C     GET THE NUMBER OF POINTS ON THE CURVE, THE CURVE SYMBOL                  DISCRVE   33
C     NUMBER, THE RECORD BEAD POINTER, THE X-COORDINATE BEAD                   DISCRVE   34
C     POINTER, AND THE Y-COORDINATE BEAD POINTER FROM THE                      DISCRVE   35
C     CURVE BEAD.                                                              DISCRVE   36
C                                                                              DISCRVE   37
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                       DISCRVE   38
      CALL DMGET(ICSYM,ICRVPTR,ISYMBOL)                                        DISCRVE   39
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                        DISCRVE   40
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                         DISCRVE   41
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                         DISCRVE   42
C                                                                              DISCRVE   43
C     IF THE NUMBER OF POINTS EXCEEDS THE PROGRAMMED LIMIT(248),               DISCRVE   44
C     DISPLAY THE FIRST 248 POINTS ONLY.                                       DISCRVE   45
C                                                                              DISCRVE   46
      IF(NPOINTS .LE. 248) GO TO 10                                            DISCRVE   47
      NPOINTS=248                                                              DISCRVE   48
      CALL DMSET(ICNPTS,ICRVPTR,NPOINTS)                                       DISCRVE   49
      CALL ERRDSM(0.,7.,62HPOINTS LIMIT EXCEEDED. THE FIRST 248 POINTS W       DISCRVE   50
     XILL BE DISPLAYED.,62,7HDISCRVE,6HINFORM)                                 DISCRVE   51
C                                                                              DISCRVE   52
```

```
      C         CONVERT THE CURVE SYMBOL NUMBER INTO A CURVE SYMBOL.          DISCRVE   53
      C                                                                       DISCRVE   54
   60    10 ISYMBOL = ISYMTAB(ISYMBOL)                                        DISCRVE   55
      C                                                                       DISCRVE   56
      C         MOVE THE CONTENTS OF THE RECORD NUMBER BEAD, THE X-COORDINATE DISCRVE   57
      C         BEAD AND THE Y-COORDINATE BEAD INTO CORE. THE RECORD NUMBER   DISCRVE   58
      C         BEAD IS A STATUS BEAD FOR THE POINTS IN THE OTHER TWO BEADS.  DISCRVE   59
   65 C              ACTIVE - THE RECORD NUMBER IS POSITIVE OR SHADDED. THE   DISCRVE   60
      C                       RECORD NUMBER IS SHADDED IF AND ONLY IF THE     DISCRVE   61
      C                       POINT HAS BEEN ADDED DURING AN EDIT.            DISCRVE   62
      C                 INACTIVE - THE RECORD NUMBER IS NEGATIVE. A POINT IS  DISCRVE   63
      C                       INACTIVE IF AND ONLY IF IT HAS BEEN DELETED     DISCRVE   64
   70 C                       DURING AN EDIT.                                 DISCRVE   65
      C                                                                       DISCRVE   66
           KOMPCOD = 700000000018 + NPOINTS * 8**6                            DISCRVE   67
           CALL DMGET(KOMPCOD,IRECPTR,NREC)                                   DISCRVE   68
           CALL DMGET(KOMPCOD,IXPTR,X)                                        DISCRVE   69
   75      CALL DMGET(KOMPCOD,IYPTR,Y)                                        DISCRVE   70
      C                                                                       DISCRVE   71
      C         LOOP THROUGH THE POINTS. IF A POINT IS INACTIVE OR NOT ON THE DISCRVE   72
      C         PLOT, SKIP IT. OTHERWISE, CONVERT THE COORDINATES TO DGUS AND DISCRVE   73
      C         PACK THE DESCRIPTION OF THE POINT INTO THE DESCRIPTION BUFFER. DISCRVE   74
   80 C                                                                       DISCRVE   75
         12 DO 20 I=1,NPOINTS                                                9JULY76    2
      287  IF(NREC(I) .LT. 0) GO TO 20                                        DISCRVE   77
           IF ((X(I).LT.XMIN).OR.(X(I).GT.XMAX))GO TO 20                      30JULY76   1
           IF ((Y(I).LT.YMIN).OR.(Y(I).GT.YMAX))GO TO 20                      30JULY76   2
   85      IX=IFIX(((X(I)-XMIN)*DGUPIN)/XINCR+XORDGU-6)                       5OCT       1
           IY=IFIX(((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU - 11.5)            DISCRVE   81
           CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                          DISCRVE   82
           CALL GUAN(ISYMBOL,1,IBUF,NBYTE,MBYTE)                              DISCRVE   83
           IF(NBYTE .LT. 312) GO TO 20                                        DISCRVE   84
   90 C                                                                       DISCRVE   85
      C         THE DESCRIPTION BUFFER IS FULL. FLUSH IT.                     DISCRVE   86
      C                                                                       DISCRVE   87
           NN=NN+1                                                            DISCRVE   88
           CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                         DISCRVE   89
   95      20 CONTINUE                                                        DISCRVE   90
      C                                                                       DISCRVE   91
      C         ALL POINTS HAVE BEEN CONSIDERED. IS THE DESCRIPTION           DISCRVE   92
      C         BUFFER EMPTY? IF NOT, FLUSH IT.                               DISCRVE   93
      C                                                                       DISCRVE   94
  100      IF(NBYTE .EQ. 0) GO TO 30                                          DISCRVE   95
           NN=NN+1                                                            DISCRVE   96
           CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                         DISCRVE   97
         30 NUMDISP = NN - INITSS + 1                                         DISCRVE   98
           RETURN                                                             DISCRVE   99
  105      END                                                                DISCRVE  100
```

```
      SUBROUTINE OCRVFIT(ITYPE,IV,X,Y,NPOINTS,C,NFP,TMIN,TMAX,TINCR,     OCRVFIT     2
     X          INITSS,NUMDISP,ICURVE)                                   28MAY76     1
C                                                                       OCRVFIT     4
C     THIS ROUTINE DISPLAYS THE CURVE FIT DEFINED BY PARAMETERS ONE     OCRVFIT     5
C     THROUGH SEVEN OVER THE INTERVAL DEFINED BY PARAMETERS EIGHT       OCRVFIT     6
C     AND NINE.                                                         OCRVFIT     7
C               SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND             13SEPT     18
C               6HAITKEN FOR A STRAIGHT FIT.                            13SEPT     19
C               SQUARES FIT AND 6HSPLINE FOR A SPLINE FIT.              OCRVFIT     9
C          IV - I - THE INDEPENDENT VARIABLE; EITHER 1HX OR 1HY.        OCRVFIT    10
C           X - I - THE ARRAY OF X-COORDINATES OF THE POINTS FOR        OCRVFIT    11
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      OCRVFIT    12
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     OCRVFIT    13
C           Y - I - THE ARRAY OF Y-COORDINATES OF THE POINTS FOR        OCRVFIT    14
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      OCRVFIT    15
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     OCRVFIT    16
C     NPOINTS - I - THE NUMBER OF POINTS WHOSE COORDINATES              OCRVFIT    17
C               ARE IN ARRAYS "X" AND "Y".                              OCRVFIT    18
C           C - I - THE ARRAY OF CURVE FIT PARAMETERS; POLYNOMIAL       OCRVFIT    19
C               COEFFICIENTS FOR A LEAST SQUARES FIT, DERIVATIVES       OCRVFIT    20
C               AT EACH POINT FOR A SPLINE FIT.                         OCRVFIT    21
C         NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF         OCRVFIT    22
C               POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT,        OCRVFIT    23
C               THE NUMBER OF POINTS FOR A SPLINE FIT.                  OCRVFIT    24
C        TMIN - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      OCRVFIT    25
C               THE DISPLAY OF THE CURVE FIT IS TO START.               OCRVFIT    26
C        TMAX - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      OCRVFIT    27
C               THE DISPLAY OF THE CURVE FIT IS TO STOP. (TMIN,TMAX)    OCRVFIT    28
C               IS THE INTERVAL OVER WHICH THE CURVE FIT IS TO BE       OCRVFIT    29
C               DISPLAYED.                                              OCRVFIT    30
C       TINCR - I - THE INCREMENT OF THE INDEPENDENT VARIABLE PER       OCRVFIT    31
C               TWENTIETH OF AN INCH. THE CURVE FIT WILL BE             OCRVFIT    32
C               APPROXIMATED BY TWENTY STRAIGHT LINES PER INCH.         OCRVFIT    33
C      INITSS - O - THE DISPLAY ADDRESSES OF THE CURVE FIT ARE          OCRVFIT    34
C               STORED IN ARRAY "IDCF". IDCF(INITSS) CONTAINS THE       OCRVFIT    35
C               DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED TO        OCRVFIT    36
C               EXHIBIT THE CURVE FIT.                                  OCRVFIT    37
C     NUMDISP - O - NUMDISP CONTAINS THE NUMBER OF DISPLAYS             OCRVFIT    38
C               REQUIRED TO EXHIBIT THE CURVE FIT.                      OCRVFIT    39
C                                                                       OCRVFIT    40
C     A 64 WORD DESCRIPTION BUFFER WILL HOLD THE DESCRIPTION BYTES      OCRVFIT    41
C     FOR 101 LINE SEGMENTS.                                            OCRVFIT    42
C          NUMBER      TYPE      BYTES                                  OCRVFIT    43
C            OF          OF        PER        TOTAL                     OCRVFIT    44
C          CALLS       CALL      CALL       BYTES                      OCRVFIT    45
C          ******      ******    *******    *******                    OCRVFIT    46
C             1        GURSET        3          3                      OCRVFIT    47
C             1        GUSEGI        4          4                      OCRVFIT    48
C           101        GUSEG         3        303                      OCRVFIT    49
C             1        GIDISP       10         10                      OCRVFIT    50
C                                                320---TOTAL            OCRVFIT    51
C     SINCE ANY CURVE FIT CAN BE DISPLAYED WITH AT MOST 100 LINE        OCRVFIT    52
C     SEGMENTS(NINE INCHES), EACH CURVE FIT CAN BE DISPLAYED WITH       OCRVFIT    53
C     AT MOST TWO BUFFERS.                                              OCRVFIT    54
C                                                                       OCRVFIT    55
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/   2
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/   2
```

```
                     X              IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),          6JULY77    2
                     X              IOINSTR(19),IODATA8(20,4,2),NCIP,IO8LOCK(20),NCIPN 1SEPT      1
      60              COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777     1
                     X    ISTR,IBUT,IMARK,ICODE,I3EAM,ISTYLE,IEOM,CPTO                 6JULY77    3
                      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                /PARMSV/   2
                      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
                     X            ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                   /CRVBEAD/  3
      65              DIMENSION TEMP(14)                                                13SEPT     20
                      DIMENSION X(NPOINTS),Y(NPOINTS),C(NFP)                            OCRVFIT    60
                C                                                                       OCRVFIT    61
                C         RECORD THE INITIAL VALUE OF THE "IOCP" SUBSCRIPT.             OCRVFIT    62
                C                                                                       OCRVFIT    63
      70              N=(ICURVE-1)*2                                                    28MAY76    2
                      INITSS = N + 1                                                    OCRVFIT    65
                C                                                                       OCRVFIT    66
                      IF(IV .EQ. 1HY) GO TO 50                                          OCRVFIT    67
                      IF(IV .EQ. 1HX) CONTINUE                                          OCRVFIT    68
      75        C                                                                       OCRVFIT    69
                C         INITIALIZE THE DESCRIPTION BUFFER TO THE FIRST POINT ON THE   OCRVFIT    70
                C         CURVE FIT. THE Y-COORDINATE OF THE POINT DEPENDS UPON THE     OCRVFIT    71
                C         TYPE OF CURVE FIT.                                            OCRVFIT    72
                C                                                                       OCRVFIT    73
      80              XX = XMIN                                                         OCRVFIT    74
                      XXINCR = XINCR                                                    OCRVFIT    75
                      XXMAX = XMAX                                                      OCRVFIT    76
                    2 IF (ITYPE.EQ.1CHLEASTSQUAR) YY=P(XX,C,NFP)                        29APR      2
                      IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)               OCRVFIT    78
      85              IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,          20SEPT     76
                     1    YY,TEMP,IER)                                                 13SEPT     22
                      IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX)      29APR      3
                     1    GO TO 5                                                      27MAY      3
                      XX=XX+XXINCR                                                      29APR      5
      90              IF (XX.GT.XXMAX) GO TO 32                                         29APR      6
                      GO TO 2                                                           29APR      7
                    5 IF (IER.EQ.2) GO TO 100                                           29APR      8
                      IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)             OCRVFIT    79
                      IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)             OCRVFIT    80
      95              NBYTE = 0                                                         OCRVFIT    81
                   10 CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                         OCRVFIT    82
                      CALL GUSEGI(IX,IY,ISTYLE,IBUF,NBYTE,MBYTE)                        OCRVFIT    83
                C                                                                       OCRVFIT    84
                C         INCREMENT THROUGH THE INTERVAL AND PACK THE DESCRIPTION BYTES  OCRVFIT    85
     100        C         OF APPROXIMATING LINE SEGMENTS INTO THE DESCRIPTION BUFFER.    OCRVFIT    86
                C                                                                       OCRVFIT    87
                   20 XX = XX + XXINCR                                                  OCRVFIT    88
                      IF(ITYPE .EQ. 1CHLEASTSQUAR) YY = P(XX,C,NFP)                     OCRVFIT    89
                      IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)               OCRVFIT    90
     105              IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,          20SEPT     77
                     1    YY,TEMP,IER)                                                 13SEPT     24
                      IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 30                            14MARCH    17
                      IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 30                            14MARCH    18
                      IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)             OCRVFIT    91
     110              IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)             OCRVFIT    92
                      CALL GUSEG(IX,IY,IBEAM)                                           OCRVFIT    93
                C                                                                       OCRVFIT    94
                C         CHECK TO SEE IF THE DESCRIPTION BUFFER IS FULL.               OCRVFIT    95
                C         IF SO, FLUSH IT. IF NOT, CONTINUE TO FILL IT.                 OCRVFIT    96
```

```
      C                                                                      14APR76   71
            IF(NBYTE .LT. 312) GO TO 30                                      DCRVFIT   97
            N = N + 1                                                        DCRVFIT   98
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                         DCRVFIT   99
            IF(XX .LT. XXMAX) GO TO 10                                       DCRVFIT  100
      C                                                                      DCRVFIT  101
      C        THE DESCRIPTION BUFFER IS NOT FULL. CHECK TO SEE IF THE CURVE DCRVFIT  102
      C         FIT IS COMPLETE. IF NOT, CONTINUE TO GENERATE THE DESCRIPTION DCRVFIT  103
      C         BYTES FOR THE CURVE FIT. IF SO, DISPLAY THE CURVE FIT.       DCRVFIT  104
      C                                                                      DCRVFIT  105
         30 IF(XX .LT. XXMAX) GO TO 20                                       DCRVFIT  106
         32 IF (NBYTE.EQ.0) GO TO 40                                         29APR     9
            N = N + 1                                                        DCRVFIT  108
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                         DCRVFIT  109
         40 NUMDISP = N - INITSS + 1                                         DCRVFIT  110
            IDCF(41)=N                                                       25MAY76   15
            RETURN                                                           DCRVFIT  111
      C                                                                      DCRVFIT  112
      C        "Y" IS THE INDEPENDENT VARIABLE. COMMENTS                     DCRVFIT  113
      C        ANALOGOUS TO THOSE ABOVE APPLY.                               DCRVFIT  114
      C                                                                      DCRVFIT  115
         50 YY = YMIN                                                        DCRVFIT  116
            YYINCR = YINCR                                                   DCRVFIT  117
            YYMAX = YMAX                                                     DCRVFIT  118
         52 IF (ITYPE.EQ.10HLEASTSQUAR) XX=P(YY,C,NFP)                       29APR    10
            IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)              DCRVFIT  120
            IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,         20SEPT    78
           1    XX,TEMP,IER)                                                 13SEPT    26
            IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX)     29APR     11
           1       GO TO 55                                                  29APR     12
            YY=YY+YYINCR                                                     29APR     13
            IF (YY.GT.YYMAX) GO TO 82                                        29APR     14
            GO TO 52                                                         29APR     15
         55 IF (IER.EQ.2) GO TO 100                                          29APR     16
            IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)            DCRVFIT  121
            IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)            DCRVFIT  122
            NBYTE = 0                                                        DCRVFIT  123
         60 CALL GUSEGI(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                        DCRVFIT  124
            CALL GUSEGI(IX,IY,ISTYLE,IBUF,NBYTE,MBYTE)                       DCRVFIT  125
         70 YY = YY + YYINCR                                                 DCRVFIT  126
            IF(ITYPE .EQ. 10HLEASTSQUAR) XX = P(YY,C,NFP)                    DCRVFIT  127
            IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)              DCRVFIT  128
            IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,         20SEPT    79
           1    XX,TEMP,IER)                                                 13SEPT    28
            IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 80                           14MARCH   21
            IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 80                           14MARCH   22
            IF (IER.EQ.2) GO TO 100                                          3MARCH    19
            IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)            DCRVFIT  129
            IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)            DCRVFIT  130
            CALL GUSEG(IX,IY,IBEAM)                                          DCRVFIT  131
            IF(NBYTE .LT. 312) GO TO 80                                      DCRVFIT  132
            N = N + 1                                                        DCRVFIT  133
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                         DCRVFIT  134
            IF(YY .LT. YYMAX) GO TO 50                                       DCRVFIT  135
         80 IF(YY .LT. YYMAX) GO TO 70                                       DCRVFIT  136
         82 IF (NBYTE.EQ.0) GO TO 90                                         29APR     17
            N = N + 1                                                        DCRVFIT  138
```

```
                  CALL GIDISP(NCCN,IBUF,NBYTE,IDCF(N),IGN)                          OCRVFIT   139
               90 NUMDISP = N - INITSS + 1                                          OCRVFIT   140
                  IDCF(41)=N                                                        25MAR76    16
175               RETURN                                                           OCRVFIT   141
               C                                                                    3MARCH     20
               C     ERROR IN AITKEN CALL - RETURN WITH NO FIT                      3MARCH     21
               C                                                                    3MARCH     22
              100 NCR=(ICURVE-1)*2+1                                                3MARCH     23
180               CALL GIPRAS (IDCF(NCR),IDCF(NCR+1))                              3MARCH     24
                  IDCF(NCR)=IDCF(NCR+1)=0                                          3MARCH     25
                  IF (ICDATAB(ICURVE,4,2).NE.0) CALL DMRLBD (ICDATAB(ICURVE,4,2))  3MARCH     26
                  ICDATAB(ICURVE,4,2)=0                                            3MARCH     27
                  CALL DMSET (ICFIT,ICDATAB(I,4,1),0)                              3MARCH     28
185               CALL ERRDUM (1.,3.5,41HINDEPENDENT VARIABLE ERROR WITH CURVE FIT, 3MARCH     29
                 1    41,7HOCRVFIT,6HINFORM)                                       3MARCH     30
                  RETURN                                                           3MARCH     31
                  END                                                              OCRVFIT   142
```

```
      FUNCTION P(Z,C,N)                                                 P          2
C                                                                       P          3
C         THIS ROUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES P          4
C         POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".          P          5
C                                                                       P          6
C         Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                P          7
C         C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.   P          8
C         N - I - THE NUMBER OF COEFFICIENTS.                           P          9
C                                                                       P          10
      DIMENSION C(N)                                                    P          11
      NDEG = N - 1                                                      P          12
      P=C(1)                                                            18JUNE76   21
      DO 10 I=2,N                                                       18JUNE76   22
      P=P+C(I)*Z**(I-1)                                                 18JUNE76   23
   10 CONTINUE                                                          P          16
      RETURN                                                            P          17
      END                                                               P          18
```

```
      1                SUBROUTINE REPLOF (NCURVE,ICRVPTR,IFITPTR,IFLAG)                    20SEPT      8
                     C                                                                   REPLOF      3
                     C     THIS ROUTINE REPLOTS THE CURVE FITS OF THE                    REPLOF      4
                     C      CURVES WHICH ARE MARKED "INACTIVE".                          REPLOF      5
      5              C                                                                   REPLOF      6
                     C                                                                   REPLOF      7
                          COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
                         X              ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
                          COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     10                  X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),       6JULY77    2
                         X               IDINSTR(19),IDDATA8(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT     1
                          COMMON /FIT9580/ IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA              REPLOF     10
                          DIMENSION X(248),Y(248),FP(248)                                    REPLOF     11
                     C                                                                       REPLOF     12
     15              C     QUERY THE CURVE AND CURVE FIT HEADS FOR THE CALL TO DCRVFIT       REPLOF     13
                     C                                                                       REPLOF     14
                          IF (IFLAG.LE.0) GO TO 20                                           20SEPT     9
                          IFITPTR=ICDATA8(NCURVE,4,2)                                        REPLOF     15
                          ICRVPTR=ICDATA8(NCURVE,4,1)                                        REPLOF     16
     20              C                                                                       REPLOF     17
                    20 CALL DMGET (IFTYPE,IFITPTR,ITYPE)                                    13SEPT     7
                       CALL DMGET (ICINDEP,ICRVPTR,IVAR)                                    REPLOF     19
                       CALL DMGET (IFIVVFP,IFITPTR,XINDFP)                                  REPLOF     20
                       CALL DMGET (IFIVVLP,IFITPTR,XINDLP)                                  REPLOF     21
     25                CALL DMGET (IFIVI,IFITPTR,XINC)                                      REPLOF     22
                     C                                                                       REPLOF     23
                          IF (ITYPE.EQ.6HAITKEN) GO TO 60                                    20SEPT     10
                          CALL DMGET (IFNPARA,IFITPTR,NFTPA)                                 REPLOF     24
                          KOMPCOD=7B0B0B0B0B0B0B68 + SHIFT(NFTPA,18)                         REPLOF     25
     30                   CALL DMGET (KOMPCOD,IFITPTR,FP)                                   REPLOF     26
                     C                                                                       REPLOF     27
                     C     IF FIT IS LEAST SQUARES CALL DCRVFIT                              REPLOF     28
                     C     SPLINE NEEDS MORE INFORMATION                                     REPLOF     29
                     C                                                                       REPLOF     30
     35                   IF (ITYPE.EQ.6HSPLINE) GO TO 50                                   REPLOF     31
                          CALL DCRVFIT (ITYPE,IVAR,XDUM,YDUM,NDUM,FP,NFTPA,XINDFP,XINDLP,   REPLOF     32
                         1     XINC,INITSS,NUMDISP,NCURVE)                                  REPLOF     33
                          GO TO 100                                                         REPLOF     34
                     C                                                                       REPLOF     35
     40              C     SPLINE FIT -- RETRIEVE EXTRA INFORMATION                          REPLOF     36
                     C                                                                       REPLOF     37
                    50 KOMPCOD=KOMPCOD+NFTPA                                                REPLOF     38
                       CALL DMGET (KOMPCOD,IFITPTR,X)                                       REPLOF     39
                       KOMPCOD=KOMPCOD+NFTPA                                                REPLOF     40
     45                CALL DMGET (KOMPCOD,IFITPTR,Y)                                       REPLOF     41
                     C                                                                       REPLOF     42
                          CALL DCRVFIT (ITYPE,IVAR,X,Y,NFTPA,FP,NFTPA,XINDFP,XINDLP,        REPLOF     43
                         1     XINC,INITSS,NUMDISP,NCURVE)                                  REPLOF     44
                          GO TO 100                                                         20SEPT     11
     50              C                                                                       20SEPT     12
                     C          AITKEN FIT -- RETRIEVE X AND Y ARRAYS                        20SEPT     13
                     C                                                                       20SEPT     14
                    60 CALL DMGET (IFNPARA,IFITPTR,NPOINTS)                                 20SEPT     15
                       CALL DMGET (IFNPARA+1,IFITPTR,NFTPA)                                 50CT       5
     55                KOMPCOD=7B0B0B0B0B0B0B7B+SHIFT(NPOINTS,18)                           20SEPT     17
                       CALL DMGET (KOMPCOD,IFITPTR,X)                                       20SEPT     18
                       CALL DMGET (KOMPCOD+NPOINTS,IFITPTR,Y)                               20SEPT     19
```

```
      C                                                                 20SEPT   20
            CALL UCRVFIT (ITYPE,IVAR,X,Y,NPOINTS,FP,NFTPA,XINCFP,XINCLP,  20SEPT   21
     60   1     XINC,INITSS,NUMDISP,NCURVE)                              20SEPT   22
      C                                                                 REPLCF   45
      C     UPDATE CURVE DISPLAY TABLE                                  REPLCF   46
      C                                                                 REPLCF   47
        100 CALL UCDATAB (NCURVE,2,SHACTIVE,INITSS,NUMDISP,IFITPTR)     REPLCF   48
     65     RETURN                                                      REPLCF   49
            END                                                         REPLCF   50
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C - - A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE       SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 8 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 8                                          SPLIN2    27
       KK = K                                                            SPLIN2    28
       GO TO 120                                                         SPLIN2    29
    8 CONTINUE                                                           SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN                                                             SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1        1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3   SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
      RETURN                                                             SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
      SUBROUTINE UCDATAB(NROW,NPAGE,ISTATUS,INITSS,NUMDISP,IBEAD)          UCDATAB    2
C                                                                          UCDATAB    3
C     THIS ROUTINE UPDATES THE CURVE DISPLAY ACCESS TABLE.                 UCDATAB    4
C                                                                          UCDATAB    5
C                  *** THE CURVE DISPLAY ACCESS TABLE ***                  UCDATAB    6
C     THE CURVE DISPLAY ACCESS TABLE IS USED TO STORE ACCESS               UCDATAB    7
C     INFORMATION ABOUT THE CURVES AND/OR CURVE FITS WHICH BELONG TO       UCDATAB    8
C     THE PLOT ""WHICH IS DISPLAYED ON THE SCREEN"". THE INFORMATION       UCDATAB    9
C          ************************************                            UCDATAB   10
C     IN THE TABLE CAN BE USED TO:                                         UCDATAB   11
C          1 - DETERMINE IF A CURVE OR A CURVE FIT FOR A CURVE IN          UCDATAB   12
C              THE PLOT IS DISPLAYED ON THE SCREEN.                        UCDATAB   13
C          2 - ERASE A CURVE OR CURVE FIT IN THE PLOT, OR                  UCDATAB   14
C          3 - RETURN TO THE CURVE BEAD OR CURVE FIT BEAD(IF IT            UCDATAB   15
C              EXISTS) OF A CURVE IN THE PLOT.                             UCDATAB   16
C     THE I-TH ROW OF ICDATAB IS INFORMATION ABOUT THE I-TH CURVE          UCDATAB   17
C     (CHRONOLOGICALLY) TO BE DISPLAYED ON THE SCREEN.                     UCDATAB   18
C               ** PAGE ONE---CURVE INFORMATION **                         UCDATAB   19
C     ICDATAB(I,1,1) IS THE STATUS OF THE I-TH CURVE. IF                   UCDATAB   20
C                    ICDATAB(I,1,1) = 6HACTIVE, THEN THE CURVE IS          UCDATAB   21
C                    CURRENTLY DISPLAYED ON THE SCREEN. IF                 UCDATAB   22
C                    ICDATAB(I,1,1) = 8HINACTIVE, THEN THE CURVE HAS       UCDATAB   23
C                    BEEN DISPLAYED ON THE SCREEN AND ERASED.              UCDATAB   24
C     ICDATAB(I,2,1) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB   25
C                    CURVE. INITSS IS EXPLAINED IN SUBROUTINE              UCDATAB   26
C                    DISCRVE.                                              UCDATAB   27
C     ICDATAB(I,3,1) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB   28
C                    EXHIBIT THE I-TH CURVE.                               UCDATAB   29
C     ICDATAB(I,4,1) IS THE POINTER TO THE CURVE BEAD OF THE I-TH          UCDATAB   30
C                    CURVE.                                                UCDATAB   31
C               ** PAGE TWO---CURVE FIT INFORMATION **                     UCDATAB   32
C     ICDATAB(I,1,2) IS THE STATUS OF THE I-TH CURVE FIT. IF               UCDATAB   33
C                    ICDATAB(I,1,2) = 4HNONE, THEN THE CURVE DOES          UCDATAB   34
C                    NOT HAVE A CURVE FIT. IF ICDATAB(I,1,2) =             UCDATAB   35
C                    6HACTIVE, THEN THE CURVE HAS A CURVE FIT AND IT       UCDATAB   36
C                    IS DISPLAYED ON THE SCREEN. IF ICDATAB(I,1,2) =       UCDATAB   37
C                    8HINACTIVE, THEN THE CURVE HAS A CURVE FIT BUT        UCDATAB   38
C                    IT IS NOT DISPLAYED ON THE SCREEN.                    UCDATAB   39
C     ICDATAB(I,2,2) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB   40
C                    CURVE FIT. INITSS IS EXPLAINED IN SUBROUTINE          UCDATAB   41
C                    DCRVFIT.                                              UCDATAB   42
C     ICDATAB(I,3,2) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB   43
C                    EXHIBIT THE I-TH CURVE FIT.                           UCDATAB   44
C     ICDATAB(I,4,2) IS THE POINTER TO THE FIT BEAD OF THE I-TH            UCDATAB   45
C                    CURVE.                                                UCDATAB   46
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),    /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),            6JULY77     2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM   1SEPT       1
      ICDATAB(NROW,1,NPAGE) = ISTATUS                                      UCDATAB   48
      ICDATAB(NROW,2,NPAGE) = INITSS                                       UCDATAB   49
      ICDATAB(NROW,3,NPAGE) = NUMDISP                                      UCDATAB   50
      ICDATAB(NROW,4,NPAGE) = IBEAD                                        UCDATAB   51
      RETURN                                                               UCDATAB   52
      END                                                                  UCDATAB   53
```

```
1                OVERLAY (4,3)                                                                  PRNTX      2
                 PROGRAM PRNTX                                                                  PRNTX      3
                                                                                                PRNTX      4
                C                                                                               PRNTX      5
                C       DUMMY TO CALL PRNTONE                                                   PRNTX      6
5               C                                                                               PRNTX      7
                COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR               /BEADS/    2
                CALL PRNTONE (IPLTPTR)                                                          PRNTX      8
                CALL AEPTRN                                                                     PRNTX      9
                END                                                                             PRNTX     10
```

```
      SUBROUTINE PRNTONE (IPLTPTR)                                      PRNTONE    2
      C                                                                 PRNTONE    3
      C        THIS ROUTINE PRINTS INFORMATION FOR THE ONE PLOT         PRNTONE    4
      C        CURRENTLY DISPLAYED                                      PRNTONE    5
      C                                                                 PRNTONE    6
      IF (IPLTPTR.EQ.0) RETURN                                          PRNTONE    7
      WRITE (6,10) IPLTPTR                                              PRNTONE    8
   10 FORMAT(1H1,50X,*LISTING OF P-M GROUP *,I10/)                      PRNTONE    9
      CALL ITRIPR (IPLTPTR)                                             PRNTONE   10
      CALL SUBPLT (6,IPLTPTR,0)                                         PRNTONE   11
      RETURN                                                            PRNTONE   12
      END                                                               PRNTONE   13
```

```
1                SUBROUTINE PRNTPLT(IPLTPTR)                                PRNTPLT     2
                 COMMON/ITRINT/ITRIM,INEXT,IPREV,N6                         /ITRINT/    2
                 EXTERNAL SUBPLT                                            PRNTPLT     4
                 IF(IPLTPTR.EQ.0)RETURN                                     PRNTPLT     5
5                WRITE(N6,10)IPLTPTR                                        PRNTPLT     6
            10   FORMAT(1H1,50X,*LISTING OF P-M GROUP *,I10/)               PRNTPLT     7
                 CALL ITRIMR(IPLTPTR)                                       PRNTPLT     8
                 CALL SPLICE(IPLTPTR,SUBPLT, 12,1,INEXT,IPREV,0)            5OCT      106
                 END                                                        PRNTPLT    10
```

```
      1                SUBROUTINE SUBPLT(I8EAD1,I8EAD,IVALU)                          PRNTPLT    13
                       COMMON/DISKT/I8D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG       30AUG       1
                 C     J=N*2**18  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1) /DISKT/      3
                       COMMON/POINTS/X(248),Y(248),NREC(248)                          /POINTS/    2
      5                COMMON /TTEMP/TEMP(10)                                         /TTEMP/     2
                       COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                             /ITRIMT/    2
                       EXTERNAL SUBCRV                                                PRNTPLT    18
                       JJ=14*2**18                                                    PRNTPLT    19
                       CALL DMGET(ICOM7+JJ,I8EAD,NREC)                                PRNTPLT    20
     10                WRITE(N6,10)I8EAD                                              PRNTPLT    21
                10     FORMAT(1H0,2X ,*LISTING OF PLOT *,I10)                         PRNTPLT    22
                       WRITE(N6,20)(NREC(I),I=1,14)                                   PRNTPLT    23
                20     FORMAT(1H ,12I8,1X,A10,1X,F10.5)                               PRNTPLT    24
                       ITITLE=NREC(6)                                                 PRNTPLT    25
     15                CALL FRNTEXT(ITITLE,10H     TITLE)                             PRNTPLT    26
                       JJ=7*2**18                                                     PRNTPLT    27
                       IF(NREC(5).EQ.0)GO TO 45                                       PRNTPLT    28
                       CALL DMGET(ICOM7+JJ,NREC(5),TEMP)                              PRNTPLT    29
                       WRITE(N6,30)NREC(5)                                            PRNTPLT    30
     20         30     FORMAT(1H0,6X ,*LISTING OF PLOT GROUP *,I10)                   PRNTPLT    31
                       WRITE(N6,40)(TEMP(I),I=1,7)                                    PRNTPLT    32
                40     FORMAT(1H ,7I8)                                                PRNTPLT    33
                       CALL FRNTEXT(TEMP(5),10H CONDITION)                            PRNTPLT    34
                45     CONTINUE                                                       PRNTPLT    35
     25                IF(NREC(7).EQ.0)GO TO 65                                       PRNTPLT    36
                       WRITE(N6,50)NREC(7)                                            PRNTPLT    37
                50     FORMAT(1H0,6X ,*LISTING OF AXIS *,I10)                         PRNTPLT
                       CALL DMGET(ICOM7+JJ,NREC(7),TEMP)                              PRNTPLT
                       WRITE(N6,60)(TEMP(I),I=1,7)                                    PRNTPLT    40
     30         60     FORMAT(1H ,4I8,1X,A10,I5,1X,A10)                               PRNTPLT    41
                       CALL FRNTEXT(TEMP(3),10H      AXIS)                            PRNTPLT    42
                65     CONTINUE                                                       PRNTPLT    43
                       IF(NREC(9).EQ.0)GO TO 75                                       PRNTPLT    44
                       WRITE(N6,70)NREC(9)                                            PRNTPLT    45
     35         70     FORMAT(1H0,6X ,*LISTING OF P-M *,I10)                          PRNTPLT    46
                       CALL DMGET(ICOM7+JJ,NREC(9),TEMP)                              PRNTPLT    47
                       WRITE(N6,40)(TEMP(I),I=1,7)                                    PRNTPLT    48
                       PARAM = TEMP(6)                                                PRNTPLT    49
                       TMULT = TEMP(7)                                                PRNTPLT    50
     40                CALL FRNTEXT(PARAM,10H PARAMETER)                              PRNTPLT    51
                       CALL FRNTEXT(TMULT,10HMULTIPLIER)                              PRNTPLT    52
                75     CONTINUE                                                       PRNTPLT    53
                       IF(NREC(8).EQ.0)GO TO 95                                       PRNTPLT    54
                       WRITE(N6,80)NREC(8)                                            PRNTPLT    55
     45         80     FORMAT(1H0,6X ,*LISTING OF AXIS SCALE *,I10)                   PRNTPLT    56
                       JJ=8*2**18                                                     T30AUG     23
                       CALL DMGET(ICOM7+JJ,NREC(8),TEMP)                              PRNTPLT    58
                       WRITE (N6,90) (TEMP(I),I=1,8)                                  T30AUG     24
                90     FORMAT(1H ,5X,4G16.8)                                          22JUNE     83
     50         95     CONTINUE                                                       PRNTPLT    61
                       IF(NREC(11).EQ.0)GO TO 99                                      PRNTPLT    62
                       CALL SPLICE2(NREC(11),SUBCRV,2,0,1,2,0)                        PRNTPLT    63
                99     CONTINUE                                                       PRNTPLT    64
                       END                                                            PRNTPLT    65
```

```
      SUBROUTINE SUBCRV(IBEAD1,IBEAD,IVALU)                                  PRNTPLT   68
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                     /ITRIMT/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/   2
      COMMON /TTEMP/TEMP(10)                                                 /TTEMP/   2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/    3
      WRITE(N6,10)IBEAD                                                      PRNTPLT   73
   10 FORMAT(1HC,11X,*LISTING OF CURVE *,I10)                                PRNTPLT   74
      JJ=12*2**18                                                            PRNTPLT   75
      CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                        PRNTPLT   76
      IF(NREC(10).NE.0)GO TO 15                                              PRNTPLT   77
      WRITE(N6,19)(NREC(I),I=1,12)                                           PRNTPLT   78
   19 FORMAT(1H ,11X,2I8,1X,A10,1X,G16.8,I5,I3,3I8,1X,I2 ,1X,I8,1X,I10)      22JUNE    84
      GO TO 16                                                               PRNTPLT   80
   15 WRITE(N6,20)(NREC(I),I=1,12)                                           PRNTPLT   81
   16 CONTINUE                                                               PRNTPLT   82
   20 FORMAT(1H ,11X,2I8,1X,A10,1X,G16.8,I5,I3,3I8,1X,A2 ,1X,I8,1X,A10)      22JUNE    85
      IR=NREC(7)                                                             PRNTPLT   84
      IX=NREC(8)                                                             PRNTPLT   85
      IY=NREC(9)                                                             PRNTPLT   86
      NPTS=NREC(5)                                                           PRNTPLT   87
      IFIT=NREC(11)                                                          PRNTPLT   88
      JJ=NPTS*2**18                                                          PRNTPLT   89
      IF(IX.EQ.0)GO TO 35                                                    PRNTPLT   90
   25 CALL DMGET(ICOM7+JJ,IR,NREC)                                           PRNTPLT   91
      CALL DMGET(ICOM7+JJ,IX,X)                                              PRNTPLT   92
      CALL DMGET(ICOM7+JJ,IY,Y)                                              PRNTPLT   93
      WRITE(N6,30)                                                           PRNTPLT   94
   30 FORMAT(1H0,18X,*REC*,17X,1HX,19X,1HY)                                  22JUNE    86
      WRITE(N6,40)(NREC(I),X(I),Y(I),I=1,NPTS)                               PRNTPLT   96
   40 FORMAT(1H ,11X,I10,10X,G16.8,4X,G16.8)                                 22JUNE    87
   35 CONTINUE                                                               PRNTPLT   98
      IF(IFIT.LE.0)RETURN                                                    PRNTPLT   99
      CALL DMGET(ICOM1+1,IFIT,FTYPE)                                         PRNTPLT  100
   35 CALL DMGET(ICOM1+2,IFIT,XO)                                            PRNTPLT  101
      CALL DMGET(ICOM1+3,IFIT,XN)                                            PRNTPLT  102
      CALL DMGET(ICOM1+4,IFIT,DELX)                                          PRNTPLT  103
      IF(FTYPE.EQ.6HAITKEN)GO TO 205                                         TOM       29
      CALL DMGET(ICOM1+5,IFIT,NP)                                            PRNTPLT  104
   40 JJ=NP*2**18+5                                                          PRNTPLT  105
      CALL DMGET(ICOM7+JJ,IFIT,X)                                            PRNTPLT  106
      WRITE(N6,100)FTYPE,XO,XN,DELX                                          PRNTPLT  107
  100 FORMAT(1H0,15X,A10,* FIT *,3F16.5)                                     PRNTPLT  108
      WRITE(N6,101)NP,(X(I),I=1,NP)                                          PRNTPLT  109
   45 101 FORMAT(1HC,20X,I5,* PARAMETERS ARE */(30X,5E16.8))                 PRNTPLT  110
      RETURN                                                                 TOM       30
  205 CALL DMGET(ICOM1+6,IFIT,IDEG)                                          TOM       31
      WRITE (N6,100) FTYPE,XO,XN,DELX                                        27MAY      8
      WRITE(N6,105)IDEG                                                      TOM       33
   50 105 FORMAT(1H ,15X,*DEGREE  *,I2)                                      TOM       34
      END                                                                    PRNTPLT  111
```

```
      SUBROUTINE SPLICE2(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)                 SPLICE2     2
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU            SPLICE2     3
C     IN WORD JMATCH OF EACH BEAD.                                              SPLICE2     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND                     SPLICE2     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                            SPLICE2     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                        SPLICE2     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                                    SPLICE2     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                        SPLICE2     9
      DATA ICOMP/0100000000000008/                                              SPLICE2    10
      NEXT=IBD1                                                                 SPLICE2    11
      IPREV=IBD1                                                                SPLICE2    12
      IF(IBD1.NE.0)GOTO10                                                       SPLICE2    13
      WRITE (6,5)                                                               CYBTTT     22
    5 FORMAT(17H IBD1 IS NULL )                                                 SPLICE2    15
      IS=2                                                                      SPLICE2    16
      RETURN                                                                    SPLICE2    17
   10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                                     SPLICE2    18
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                       SPLICE2    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                               SPLICE2    20
      IF(NEXTAD.EQ.0)GOTO12                                                     SPLICE2    21
      IPREV=NEXT                                                                SPLICE2    22
      NEXT=NEXTAD                                                               SPLICE2    23
      GO TO 10                                                                  SPLICE2    24
   12 WRITE (6,6) IVALU                                                         CYBTTT     23
    6 FORMAT(19H NO MATCH FOR IVALU ,020)                                       SPLICE2    26
      IS=2                                                                      SPLICE2    27
      RETURN                                                                    SPLICE2    28
   15 CONTINUE                                                                  SPLICE2    29
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL                 SPLICE2    30
      IF(NEXT.EQ.IBD1)GO TO 25                                                  SPLICE2    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE             SPLICE2    32
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                           SPLICE2    33
      IF(NEXTAD.EQ.0)GOTO22                                                     SPLICE2    34
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                       SPLICE2    35
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                          SPLICE2    36
   22 CONTINUE                                                                  SPLICE2    37
      CALL SUBR(IBD1,NEXT,IVALU)                                                SPLICE2    38
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                              SPLICE2    39
      RETURN                                                                    SPLICE2    40
   25 CALL SUBR(IBD1,IBD1,IVALU)                                                SPLICE2    41
      IF(NEXTAD.EQ.0)GO TO 40                                                   SPLICE2    42
      NEXT=NEXTAD                                                               SPLICE2    43
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                       SPLICE2    44
      CALL SUBR(IBD1,NEXT,IVALU)                                                SPLICE2    45
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                              SPLICE2    46
      IF(NEXTAD.EQ.0)GO TO 40                                                   SPLICE2    47
      NEXT=NEXTAD                                                               SPLICE2    48
      GO TO 30                                                                  SPLICE2    49
   40 CONTINUE                                                                  SPLICE2    50
      IF(IS.EQ.1)CALL DMRLBD(IBD1)                                              SPLICE2    51
      END                                                                       SPLICE2    52
```

```
      SUBROUTINE ITRIMR(IBEAD)                                          ITRIMR     2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**16  CMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      COMMON/OPTION/ICPTION,ISHINE                                      LASTBUX    1
      CALL CMGET(ICOM1+2,IBEAD,ITRIM)                                   ITRIMR     5
      IF(ITRIM.EQ.0)GO TO 10                                            ITRIMR     6
      IF (ICPTION.NE.4HTRIM) GO TO 10                                   3AUG      16
      INEXT=2                                                           ITRIMR     7
10    IPREV=4                                                           ITRIMR     8
C                                                                       ITRIMR     9
C     TRIM MODE                                                         ITRIMR    10
      RETURN                                                            ITRIMR    11
 10   CONTINUE                                                          ITRIMR    12
C     CROSSSLOT MODE                                                    ITRIMR    13
      INEXT=1                                                           ITRIMR    14
      IPREV=3                                                           ITRIMR    15
      END                                                               ITRIMR    16
```

```
      SUBROUTINE PRNTEXT(IBEAD,TNAME)                                   PRNTEXT    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      COMMON /TTEMP/TEMP(10)                                            /TTEMP/    2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/   2
      INTEGER TEMP                                                      PRNTEXT    6
      IBD=IBEAD                                                         PRNTEXT    7
      JJ=10*2**18                                                       PRNTEXT    8
   99 IF(IBD.LE.0)RETURN                                                PRNTEXT    9
      CALL DMGET(ICOM7+JJ,IBD,TEMP)                                     PRNTEXT   10
      NW=TEMP(2)/10                                                     PRNTEXT   11
      IF(( TEMP(2)-10*NW).GT.0)NW=NW+1                                  PRNTEXT   12
      NW=NW+2                                                           PRNTEXT   13
      WRITE(N6,100)TNAME,IBD,(TEMP(I),I=1,NW)                          PRNTEXT   14
  100 FORMAT(1H0,30X,A10,* TEXT *,I10/1H ,15X,2I10,1X,8A10)             PRNTEXT   15
      IBD=TEMP(1)                                                       PRNTEXT   16
      GO TO 99                                                          PRNTEXT   17
      END                                                               PRNTEXT   18
```

```
      1              OVERLAY(5,0)                                                    EDIT        2
                     PROGRAM EDIT                                                    EDIT        3
                C                                                                    EDIT        4
                C        THIS ROUTINE SETS THE SCREEN AND MANAGES THE CALL OF THE    EDIT        5
      5         C        DELETE POINT, ADD POINT, DELETE CURVE, AND RETURN TO        EDIT        6
                C        QUARTERBACK ROUTINES. PRIMARY INPUT TO THE EDIT "OVERLAY"   EDIT        7
                C        IS THE CURVE DISPLAY ACCESS TABLE.                          EDIT        8
                C                                                                    EDIT        9
                     COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     10            X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),        6JULY77     2
                   X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
                     COMMON/EDITFX/IDDCSTR                                           INHALE      37
                     COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                CY8777      1
                    X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO               6JULY77     3
     15         C                                                                    EDIT        12
                C        CHANGE THE PICK TYPE OF EACH OF THE CURVE DESCRIPTIONS FROM  EDIT        13
                C        IGNORE TO STRING, DISPLAY THE EDIT MENU AND INSTRUCTIONS, AND EDIT       14
                C        WAIT FOR A BUTTON PICK FROM THE FOLLOWING MENU:              EDIT        15
                C              21 - DELETE POINT                                      EDIT        16
     20         C              22 - ADD POINT                                         EDIT        17
                C              23 - DELETE CURVES                                     EDIT        18
                C              25 - DELETE PLOT                                       3NOV        5
                C              24 - RETURN                                            EDIT        19
                C             254 - STOP                                              14APR76     38
     25         C                                                                    EDIT        21
305             10 CALL CPTYPE(IDCD,20,1,NCIP,ISTR)                                 EDIT        22
                   CALL GICLR(NCON)                                                 EDIT        23
                   CALL EDITM                                                       EDIT        24
                   CALL EDITT                                                       EDIT        25
     30            CALL ORIGIN(-5.,-3.)                                             5MAY76      14
                20 CALL GIBUT(0,NCON,IDOT,IDDCBUT)                                  EDIT        26
                   IF(IDDCBUT .EQ. 254) CALL STOP                                   14APR76     39
                   IF(IDDCBUT .EQ. 24) GO TO 30                                     5MAY76      15
                   IF (IDDCBUT.EQ.25) GO TO 40                                      3NOV        6
     35         C                                                                    EDIT        29
                C        CHECK FOR THE EXISTENCE OF A STRING PICK.                   EDIT        30
                C                                                                    EDIT        31
                   N=1                                                              EDIT        32
                   CALL GIFSID(NCON,N,ICCT,IDDCSTR)                                 EDIT        33
     40            IF(N .EQ. 1) GO TO 30                                            EDIT        34
                   CALL ERROFM(1.7,7.0,61HYOU DID NOT PICK A CURVE DESCRIPTION BEFORE 5MAY76    16
                  X PICKING A BUTTON.,61,4HEDIT,6HINFORM)                           EDIT        36
                   GO TO 20                                                         EDIT        37
                C                                                                    EDIT        38
     45         C        THERE IS AT LEAST ONE STRING PICK.                          EDIT        39
                C             1 - CHANGE THE PICK TYPE OF EACH OF THE CURVE          EDIT        40
                C                 DESCRIPTIONS BACK TO IGNORE,                       EDIT        41
                C             2 - ERASE THE EDIT MENU AND INSTRUCTIONS, AND          EDIT        42
                C             3 - IDENTIFY THE BUTTON PICK.                          EDIT        43
     50         C                                                                    EDIT        44
                30 CALL CPTYPE(IDCD,20,1,NCIP,IGN)                                  EDIT        45
                   CALL TERASE(IDMENU,IDMENU(13))                                   EDIT        46
                   CALL TERASE(IDINSTR,IDINSTR(19))                                 EDIT        47
                   IF (IDDCBUT .EQ. 21) CALL AETSKC (6HDELPNT)                      INHALE      38
     55            IF (IDDCBUT .EQ. 22) CALL AETSKC (6HADDPNT)                      INHALE      39
                   IF (IDDCBUT .EQ. 23) CALL AETSKC (7HDELCRVE)                     INHALE      40
                   IF(IDDCBUT .EQ. 24) CALL AETSKC(7HQTRBACK)                       5MAY76      17
```

```
                  CALL AETSKC (7HREPLOTX)                                    INHALE     41
                  GO TC 10                                                   2NOV       44
60              40 CALL ANNOTA (10.,-2.75,12HARE YOU SURE,12,IAC1,IGN,-0,-0,-0)  16NOV     4
                  CALL ANNOTA (9.9,-3.,3HYES,3,IAC2,IBUT,99,-0,-0)           16NOV      5
                  CALL ANNOTA (11.1,-3.,2HNO,2,IAC3,IBUT,100,-0,-0)          16NOV      6
                  CALL GIBUT (0,NCON,ICCT,IDDCBUT)                           3NOV       8
                  IF (IDDCBUT.EC.99) GO TO 50                                3NOV       9
65                IF (IDDCBUT.EQ.25+) CALL STOP                              3NOV       10
                  CALL GIERAS (IAC1,IAC2,IAC3)                               2NOV       48
                  GO TC 20                                                   2NOV       49
                50 CALL GIERAS (IAC1,IAC2,IAC3)                              2NOV       50
                  CALL LPTYPE (IOCO,20,1,NCIP,IGN)                           3NOV       13
70                CALL TERASE (IDMENU,IDMENU(13))                            3NOV       14
                  CALL TERASE (IDINSTR,IDINSTR(19))                          3NOV       15
                  CALL AETSKC (7HCELPLOT)                                    INHALE     42
                  CALL AETSKC (7HQTRBACK)                                    3NOV       17
                  END                                                        EDIT       52
```

```
      SUBROUTINE EDITM                                                  EDITM     2
C                                                                       EDITM     3
C         THIS ROUTINE DISPLAYS THE EDIT MENU.                          EDITM     4
C                                                                       EDITM     5
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),         6JULY77   2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT    1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBER77   1
     X    ISTP,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77   3
      CALL ORIGIN(0.,-5.)                                               EDITM     8
      CALL SQUARE (0.,-4.,3.,4.,IDMENU(1),IGN,-0)                       3NOV      25
      CALL ANNOTA(1.26,.29,4HMENU,4,IDMENU(2),IGN,-0,-0,-0)             EDITM     10
      CALL ANNOTA(.78,-.69,12HDELETE POINT,12,IDMENU(3),IBUT,21,-0,-0)  EDITM     11
      CALL ANNOTA(.78,-1.19,9HADD POINT,9,IDMENU(4),IBUT,22,-0,-0)      EDITM     12
      CALL ANNOTA(.78,-1.69,13HDELETE CURVES,13,IDMENU(5),IBUT,23,-0,-0) EDITM    13
      CALL ANNOTA(.78,-2.19,6HRETURN,6,IDMENU(6),IBUT,24,-0,-0)         EDITM     14
      CALL ANNOTA (.78,-2.19,11HDELETE PLOT,11,IDMENU(7),IBUT,25,-0,-0) 3NOV      26
      IDMENU(13)=7                                                      3NOV      27
      RETURN                                                            EDITM     16
      END                                                               EDITM     17
```

```
      SUBROUTINE EDITI                                                  EDIT1     2
C                                                                       EDITI     3
C     THIS ROUTINE DISPLAYS THE INSTRUCTIONS                            EDITI     4
C     WHICH GO WITH THE EDIT MENU.                                      EDITI     5
C                                                                       EDITI     6
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),         6JULY77   2
     Y               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPN 1SEPT    1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777    1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77   3
      CALL ORIGIN(-5.5,-5.)                                             EDITI     9
      CALL SQUARE (0.,-4.75,4.5,4.75,IDINSTR(1),IGN,-0)                 3NOV      18
      CALL ANNOTA(1.53,.25,12HINSTRUCTIONS,12,IDINSTR(2),IGN,-0,-0,-0)  EDITI     11
      CALL ANNOTA( .75,-.5,25HDELETE POINT OR ADD POINT,25,IDINSTR(3),  EDITI     12
     X             IGN,-0,-0,-0)                                        EDITI     13
      CALL ANNOTA(.45,-.75,29H1.SINGLE PICK THE DESCRIPTION,29,         EDITI     14
     X             IDINSTR(4),IGN,-0,-0,-0)                             EDITI     15
      CALL ANNOTA(.69,-1.,27HOF THE CURVE TO BE ALTERED.,27,IDINSTR(5), EDITI     16
     X             IGN,-0,-0,-0)                                        EDITI     17
      CALL ANNOTA(.45,-1.25,26H2.PICK DELETE POINT OR ADD,26,IDINSTR(6), EDITI     18
     Y             IGN,-0,-0,-0)                                        EDITI     19
      CALL ANNOTA(.69,-1.5,6HPOINT.,6,IDINSTR(7),IGN,-0,-0,-0)          EDITI     20
      CALL ANNOTA(1.47,-2.,13HDELETE CURVES,13,IDINSTR(8),IGN,-0,-0,-0) EDITI     21
      CALL ANNOTA(.45,-2.25,29H1.STRING PICK THE DESCRIPTION,29,        EDITI     22
     X             IDINSTR(9),IGN,-0,-0,-0)                             EDITI     23
      CALL ANNOTA(.69,-2.5,28HOF THE CURVES TO BE DELETED.,28,          EDITI     24
     X             IDINSTR(10),IGN,-0,-0,-0)                            EDITI     25
      CALL ANNOTA(.45,-2.75,21H2.PICK DELETE CURVES.,21,IDINSTR(11),    EDITI     26
     X             IGN,-0,-0,-0)                                        EDITI     27
      CALL ANNOTA (1.47,-3.,11HDELETE PLOT,11,IDINSTR(12),IGN,-0,-0,-0) 3NOV      19
      CALL ANNOTA (.45,-3.25,28HTO DELETE ALL CURVES ON PLOT, 28,       3NOV      20
     1    IDINSTR(13),IGN,-0,-0,-0)                                     3NOV      21
      CALL ANNOTA (.69,-3.5,17HPICK DELETE PLOT.,17,IDINSTR(14),IGN,-0, 3NOV      22
     1   -0,-0)                                                         3NOV      23
      IDINSTR(19)=14                                                    3NOV      24
      RETURN                                                            EDITI     29
      END                                                               EDITI     30
```

```
1                SUBROUTINE CPTYPE(IDDAD,L,M,N,NEWTYPE)                                  CPTYPE     2
                 C                                                                       CPTYPE     3
                 C       THIS ROUTINE CHANGES THE PICK TYPE OF SOME OF THE DISPLAYS      CPTYPE     4
                 C       WHOSE ADDRESSES ARE STORED IN ARRAY IDDAD. THE PICK TYPE OF     CPTYPE     5
5                C       DISPLAYS IDDAD(M) THROUGH IDDAD(N) IS CHANGED FROM ITS CURRENT CPTYPE     6
                 C       TYPE TO NEWTYPE. "L" IS THE LENGTH OF THE ARRAY IDDAD.          CPTYPE     7
                 C                                                                       CPTYPE     8
                      DIMENSION IDDAD(L)                                                 CPTYPE     9
                      DO 10 I=M,N                                                        CPTYPE    10
10                    CALL GIMOVE(-0,-0,-0,IDDAD(I),NEWTYPE)                             CPTYPE    11
                   10 CONTINUE                                                           CPTYPE    12
                      RETURN                                                             CPTYPE    13
                      END                                                                CPTYPE    14
```

```
      SUBROUTINE UCDATAB(NROW,NPAGE,ISTATUS,INITSS,NUMDISP,IBEAD)          UCDATAB    2
C                                                                          UCDATAB    3
C     THIS ROUTINE UPDATES THE CURVE DISPLAY ACCESS TABLE.                 UCDATAB    4
C                                                                          UCDATAB    5
C               *** THE CURVE DISPLAY ACCESS TABLE ***                     UCDATAB    6
C     THE CURVE DISPLAY ACCESS TABLE IS USED TO STORE ACCESS               UCDATAB    7
C     INFORMATION ABOUT THE CURVES AND/OR CURVE FITS WHICH BELONG TO       UCDATAB    8
C     THE PLOT ""WHICH IS DISPLAYED ON THE SCREEN"". THE INFORMATION       UCDATAB    9
C               ************************************                       UCDATAB   10
C     IN THE TABLE CAN BE USED TO:                                         UCDATAB   11
C          1 - DETERMINE IF A CURVE OR A CURVE FIT FOR A CURVE IN          UCDATAB   12
C              THE PLOT IS DISPLAYED ON THE SCREEN.                        UCDATAB   13
C          2 - ERASE A CURVE OR CURVE FIT IN THE PLOT, OR                  UCDATAB   14
C          3 - RETURN TO THE CURVE BEAD OR CURVE FIT BEAD(IF IT            UCDATAB   15
C              EXISTS) OF A CURVE IN THE PLOT.                             UCDATAB   16
C     THE I-TH ROW OF ICDATAB IS INFORMATION ABOUT THE I-TH CURVE          UCDATAB   17
C     (CHRONOLOGICALLY) TO BE DISPLAYED ON THE SCREEN.                     UCDATAB   18
C                  ** PAGE ONE---CURVE INFORMATION **                      UCDATAB   19
C     ICDATAB(I,1,1) IS THE STATUS OF THE I-TH CURVE. IF                   UCDATAB   20
C                    ICDATAB(I,1,1) = 8HACTIVE, THEN THE CURVE IS          UCDATAB   21
C                    CURRENTLY DISPLAYED ON THE SCREEN. IF                 UCDATAB   22
C                    ICDATAB(I,1,1) = 8HINACTIVE, THEN THE CURVE HAS       UCDATAB   23
C                    BEEN DISPLAYED ON THE SCREEN AND ERASED.              UCDATAB   24
C     ICDATAB(I,2,1) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB   25
C                    CURVE. INITSS IS EXPLAINED IN SUBROUTINE              UCDATAB   26
C                    DISCRVE.                                              UCDATAB   27
C     ICDATAB(I,3,1) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB   28
C                    EXHIBIT THE I-TH CURVE.                               UCDATAB   29
C     ICDATAB(I,4,1) IS THE POINTER TO THE CURVE BEAD OF THE I-TH          UCDATAB   30
C                    CURVE.                                                UCDATAB   31
C                  ** PAGE TWO---CURVE FIT INFORMATION **                  UCDATAB   32
C     ICDATAB(I,1,2) IS THE STATUS OF THE I-TH CURVE FIT. IF               UCDATAB   33
C                    ICDATAB(I,1,2) = 4HNONE, THEN THE CURVE DOES          UCDATAB   34
C                    NOT HAVE A CURVE FIT. IF ICDATAB(I,1,2) =             UCDATAB   35
C                    8HACTIVE, THEN THE CURVE HAS A CURVE FIT AND IT       UCDATAB   36
C                    IS DISPLAYED ON THE SCREEN. IF ICDATAB(I,1,2) =       UCDATAB   37
C                    8HINACTIVE, THEN THE CURVE HAS A CURVE FIT BUT        UCDATAB   38
C                    IT IS NOT DISPLAYED ON THE SCREEN.                    UCDATAB   39
C     ICDATAB(I,2,2) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB   40
C                    CURVE FIT. INITSS IS EXPLAINED IN SUBROUTINE          UCDATAB   41
C                    DCRVFIT.                                              UCDATAB   42
C     ICDATAB(I,3,2) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB   43
C          .         EXHIBIT THE I-TH CURVE FIT.                           UCDATAB   44
C     ICDATAB(I,4,2) IS THE POINTER TO THE FIT BEAD OF THE I-TH            UCDATAB   45
C                    CURVE.                                                UCDATAB   46
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IOSPEC(19),IOCF(41),    /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IOCO(23),IOMENU(20),            6JULY77    2
     X               IOINSTR(19),ICDATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM   1SEPT      1
      ICDATAB(NROW,1,NPAGE) = ISTATUS                                      UCDATAB   48
      ICDATAB(NROW,2,NPAGE) = INITSS                                       UCDATAB   49
      ICDATAB(NROW,3,NPAGE) = NUMDISP                                      UCDATAB   50
      ICDATAB(NROW,4,NPAGE) = IBEAD                                        UCDATAB   51
      RETURN                                                               UCDATAB   52
      END                                                                  UCDATAB   53
```

```
      1                 OVERLAY (5,4)                                                  INHALE     43
                        PROGRAM DELPLOT                                                INHALE     44
                C                                                                      DELPLOT     3
                C         THIS ROUTINE DELETES THE ENTIRE PLOT FROM THE DATA FILE,      DELPLOT     4
      5         C         DISABLES THE TRIM OPTION AND ERASES THE CURVES AND CURVE     DELPLOT     5
                C         FITS.  SUBSEQUENT PASSES ON THE DATA FILE WILL SKIP THIS PLOT DELPLOT     6
                C                                                                      DELPLOT     7
                          COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/     2
                          COMMON/DISPLAY/IOSTOP,IODAD(80),NM,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/    2
     10                 X                IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),     6JULY77    2
                        X                IOINSTR(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT     1
                          COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE, /PLTBEAD/     2
                        X                IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT, /PLTBEAD/     3
                        X                IPFMN,IPFMV                                  /PLTBEAD/     4
     15                 COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC, /CRVBEAD/     2
                        X                ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT           /CRVBEAD/     3
                          COMMON/PMBEAD/IPMNEXT,IPMPREV,IPMPLST,IPMMLST,IPMPLOT,IPMPTXT, /PMBEAD/     2
                        X                IPMMTXT                                      /PMBEAD/     3
                          COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS, /PGBEAD/     2
     20                 X                IPGLCA                                       /PGBEAD/     3
                          COMMON/AXBEAD/IAXNEXT,IAXPREV,IAXATXT,IAXPM,IAXXNME,IAXNYAX,IAXY1, /AXBEAD/     2
                        X                IAXY2,IAXY3                                  /AXBEAD/     3
                C                                                                      DELPLOT    13
                C         FIRST DELETE CURVES AND CURVE FITS FROM PLOT DISPLAY         DELPLOT    14
     25         C                                                                      DELPLOT    15
                          NLOC=NCIPM*4                                                 DELPLOT    16
                          CALL TERASE (IODAD,NLOC)                                     DELPLOT    17
                          NM=0                                                         DELPLOT    18
                          CALL TERASE (IODCF,NLOC)                                     DELPLOT    19
     30                   IDCF(41)=0                                                   DELPLOT    20
                          CALL TERASE (IOCO,NCIP)                                      DELPLOT    21
                          CALL DMGET (IPPG,IPLTPTR,IPGPTR)                             2NOV       53
                          CALL DMGET (IPGAXIS,IPGPTR,IAXPTR)                           2NOV       54
                          CALL DMGET (IAXPM,IAXPTR,IPMPTR)                             2NOV       55
     35         C                                                                      DELPLOT    22
                C         NOW FIX THE DATA FILE                                        DELPLOT    23
                C                                                                      DELPLOT    24
                          CALL DMGET (IPNEXTC,IPLTPTR,NEXTCP)                          DELPLOT    25
                          CALL DMGET (IPNEXTT,IPLTPTR,NEXTTP)                          DELPLOT    26
     40                   CALL DMGET (IPPREVC,IPLTPTR,IPREVCP)                         DELPLOT    27
                          CALL DMGET (IPPREVT,IPLTPTR,IPREVTP)                         DELPLOT    28
                          IF (NEXTCP.NE.0) CALL DMSET (IPPREVC,NEXTCP,IPREVCP)         DELPLOT    29
                          IF (NEXTTP.NE.0) CALL DMSET (IPPREVT,NEXTTP,IPREVTP)         DELPLOT    30
                          IF (IPREVCP.NE.0) CALL DMSET (IPNEXTC,IPREVCP,NEXTCP)        DELPLOT    31
     45                   IF (IPREVTP.NE.0) CALL DMSET (IPNEXTT,IPREVTP,NEXTTP)        DELPLOT    32
                C                                                                      DELPLOT    33
                C         FIND AND DELETE CURVE AND CURVE FIT BEADS                    DELPLOT    34
                C                                                                      DELPLOT    35
                          CALL DMGET (IPCURVE,IPLTPTR,ICURVE)                          DELPLOT    36
     50                40 CALL DMGET (ICFIT,ICURVE,IFIT)                               DELPLOT    37
                          IF (IFIT.NE.0) CALL DMRLSO (IFIT)                            DELPLOT    38
                          CALL DMGET (ICNEXT,ICURVE,ICNXT)                             DELPLOT    39
                          CALL DMRLSO (ICURVE)                                         DELPLOT    40
                          IF (ICNXT.EQ.0) GO TO 60                                     DELPLOT    41
     55                   ICURVE=ICNXT                                                 DELPLOT    42
                          GO TO 40                                                     DELPLOT    43
                C                                                                      DELPLOT    44
```

```
      C        END OF CURVES.  SET FIRST "NEXT TRIM PLOT POINTER"          DELPLOT    45
      C        TO ZERO TO INHIBIT TRIM OPTION                              DELPLOT    46
60    C                                                                    DELPLOT    47
   60 IF (IPREVCP.NE.0) GO TO 70                                           16NOV       7
      CALL DMSET (IPNEXTT,NEXTTP,0)                                        DELPLOT    49
      CALL DMSET (IPMPLOT,IPMPTR,NEXTCP)                                   2NOV       56
      GO TO 100                                                            DELPLOT    51
65    C                                                                    DELPLOT    52
   70 CALL DMGET (IPPREVC,IPREVCP,IPREVCX)                                 16NOV       8
      IF (IPREVCX.EQ.0) GO TO 80                                           16NOV       9
      IPREVCP=IPREVCX                                                      16NOV      10
      GO TO 70                                                             16NOV      11
70    80 CALL DMSET (IPNEXTT,IPREVCP,0)                                    16NOV      12
      CALL DMSET (IPMPLOT,IPMPTR,IPREVCP)                                  16NOV      13
  100 CALL ABRTRN                                                          INHALE     45
      END                                                                  DELPLOT    61
```

```
      1                 OVERLAY (5,1)                                                    INHALE     46
                        PROGRAM ADDPNT                                                   INHALE     47
                C                                                                        ADDPNT      3
                C   THIS ROUTINE ADDS POINTS TO A SELECTED CURVE                       ADDPNT      4
                C                                                                        ADDPNT      5
                      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/   2
                      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,   /CRVBEAD/   2
                     X              ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                  /CRVBEAD/   3
                      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/   2
     10               X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
                      X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
                      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYB777     1
                     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77    3
                      COMMON/PARMSV/XGRDGU,YGRDGU,DGUPIN                                 /PARMSV/   2
     15               COMMON/POINTS/X(248),Y(248),NPEC(248)                              /POINTS/   2
                      COMMON /SYMTAB/ ISYMTAB(20)                                        /SYMTAB/   2
                      COMMON/EDITFX/IDDCSTR                                              INHALE     48
                      DIMENSION IDPT(10), XD(10), YD(10)                                 ADDPNT     13
                      NPAD=0                                                             ADDPNT     14
     20               C                                                                  ADDPNT     15
                C         IDDCSTR INDIRECTLY IDENTIFIES THE CURVE TO WHICH POINTS ARE    ADDPNT     16
                C         TO BE ADDED.  IDDCSTR IS THE NUMBER OF A ROW IN THE CURVE      ADDPNT     17
                C         DISPLAY ACCESS TABLE.  THAT ROW CORRESPONDS TO A CURVE ON THE  ADDPNT     18
                C         SCREEN.                                                        ADDPNT     19
     25         C                                                                        ADDPNT     20
                C         IF MORE THAN ONE CURVE WAS PICKED, THEN THE CURVE TO WHICH     ADDPNT     21
                C         POINTS WILL BE ADDED IS THE LAST ONE.  WHEN CONTROL PASSES     ADDPNT     22
                C         TO THIS PROGRAM, IDDCSTR IS THE IDDC OF THE FIRST CURVE        ADDPNT     23
                C         PICKED.  RETRIEVE THE IDDC OF THE LAST CURVE PICKED.           ADDPNT     24
     30         C                                                                        ADDPNT     25
               10 NROW=IDDCSTR                                                           ADDPNT     26
                  N=1                                                                    ADDPNT     27
                  CALL GIFSID (NCON,N,IDDT,IDDCSTR)                                      ADDPNT     28
                  IF (N.EQ.1) GO TO 10                                                   ADDPNT     29
     35           C                                                                      ADDPNT     30
                  C         1 - ERASE ALL CURVES, CURVE FITS AND DESCRIPTIONS BUT THE    ADDPNT     31
                  C             SELECTED CURVE AND DESCRIPTION.                          ADDPNT     32
                  C         2 - IN THE CURVE DISPLAY ACCESS TABLE CHANGE THE STATUS OF EAC ADDPNT     33
                  C             ERASED CURVE AND CURVE FIT FROM ACTIVE TO INACTIVE.      ADDPNT     34
     40           C         3 - DISPLAY THE INSTRUCTIONS AND MENU.                       ADDPNT     35
                  C         4 - WAIT FOR THE BUTTON PICK.                                ADDPNT     36
                  C                                                                      ADDPNT     37
                  DO 30 I=1,NCIP                                                         ADDPNT     38
                  IF (I.EQ.NROW) GO TO 20                                                ADDPNT     39
     45           ICDATAB(I,1,1) = 8HINACTIVE                                            ADDPNT     40
                  CALL TERASE (IDCDAD(ICDATAB(I,2,1)),ICDATAB(I,3,1))                    ADDPNT     41
                  CALL GIERAS (IDDC(I))                                                  ADDPNT     42
                C                                                                        ADDPNT     43
                C         CHECK TO SEE IF THERE IS AN ACTIVE CURVE FIT                   ADDPNT     44
     50         C                                                                        ADDPNT     45
               20 IF (ICDATAB(I,4,2).LE.0) GO TO 30                                      ADDPNT     46
                  ICDATAB(I,1,2)=8HINACTIVE                                              ADDPNT     47
                  CALL TERASE (IDCF(ICDATAB(I,2,2)),ICDATAB(I,3,2))                      ADDPNT     48
               30 CONTINUE                                                               ADDPNT     49
     55        35 CALL ADINST                                                           ADDPNT     50
                  CALL ADMENU                                                            ADDPNT     51
                C                                                                        ADDPNT     52
```

```
      C          WAIT FOR A BUTTON PICK FROM -                                ADDPNT     53
      C               40 - KEYBOARD ENTRY                                     ADDPNT     54
   60 C               41 - TRACKING CROSS                                     ADDPNT     55
      C                                                                       ADDPNT     56
         CALL ORIGIN (-5.,-3.)                                                ADDPNT     57
      40 CALL GIBUT (0,NCON,IDCT,IDDC)                                        ADDPNT     58
         IF (IDDC.EQ.40) GO TO 50                                             ADDPNT     59
   65    IF (IDDC.EQ.41) GO TO 60                                             ADDPNT     60
         IF (IDDC.EQ.254) CALL STOP                                           ADDPNT     61
         GO TO 40                                                             ADDPNT     62
      C                                                                       ADDPNT     63
      C          SELECTED KEYBOARD ENTRY.  ERASE THE INSTRUCTIONS AND MENU    ADDPNT     64
   70 C          AND CALL ROUTINE ADDKEY                                      ADDPNT     65
      C                                                                       ADDPNT     66
      50 CALL ORIGIN (-5.,-3.)                                                ADDPNT     67
         CALL TERASE (IDINSTR,IDINSTR(19))                                    ADDPNT     68
         CALL TERASE (IDMENU,IDMENU(13))                                      ADDPNT     69
   75    CALL ADDKEY (IXAD,IYAD,XV,YV)                                        ADDPNT     70
         GO TO 62                                                             ADDPNT     71
      C                                                                       ADDPNT     72
      C          SELECTED TRACKING CROSS                                      ADDPNT     73
      C                                                                       ADDPNT     74
   80 60 CALL ORIGIN (-5.,-3.)                                                ADDPNT     75
         CALL TERASE (IDINSTR,IDINSTR(19))                                    ADDPNT     76
         CALL TERASE (IDMENU,IDMENU(13))                                      ADDPNT     77
         CALL ADDTKC (IXAD,IYAD,XV,YV)                                        ADDPNT     78
      C                                                                       ADDPNT     79
   85 C          DISPLAY POINT AND ASK FOR CONFIRMATION                       ADDPNT     80
      C                                                                       ADDPNT     81
      62 NPAD=NPAD+1                                                          ADDPNT     82
          XD(NPAD)=XV                                                         ADDPNT     83
          YD(NPAD)=YV                                                         ADDPNT     84
   90     ICRVPTR=ICDATAB(IDDCSTR,4,1)                                        ADDPNT     85
          CALL GURSET (IXAD-6 ,IYAD-11,ICODE,IBUF,NBYTE,MBYTE)                SOCT       33
          CALL DMGET (ICSYM,ICRVPTR,ISYM)                                     ADDPNT     87
          CALL GUAN (ISYMTAB(ISYM),1,IBUF,NBYTE,MBYTE)                        ADDPNT     88
          CALL GIDISP (NCON,IBUF,NBYTE,IOPT(NPAD))                            ADDPNT     89
   95 C                                                                       ADDPNT     90
      C          DISPLAY THE SELECTION MENU                                   ADDPNT     91
      C                                                                       ADDPNT     92
         CALL ADINST2                                                         ADDPNT     93
      C                                                                       ADDPNT     94
  100 C          WAIT FOR A BUTTON PICK -                                     ADDPNT     95
      C               1 - CONTINUE ADDING                                     ADDPNT     96
      C               2 - RESTORE CURVE                                       ADDPNT     97
      C               3 - SELECT NEXT CURVE                                   ADDPNT     98
      C                                                                       ADDPNT     99
  105    65 CALL GIBUT (0,NCON,IDCT,IDDC)                                     ADDPNT    100
         IF (IDDC.EQ.1) GO TO 70                                              ADDPNT    101
         IF (IDDC.EQ.2) GO TO 80                                              ADDPNT    102
         IF (IDDC.EQ.3) GO TO 90                                              ADDPNT    103
         IF (IDDC.EQ.254) CALL STOP                                           ADDPNT    104
  110 C                                                                       ADDPNT    105
      70 CALL TERASE (IDINSTR,IDINSTR(19))                                    ADDPNT    106
         GO TO 35                                                             ADDPNT    107
      C                                                                       ADDPNT    108
      90 CALL TERASE (IOPT,NPAD)                                              ADDPNT    109
```

```
115               NPAD=0                                                    SOCT      34
                  GO TO 65                                                  ADDPNT   110
                C                                                           ADDPNT   111
                  90 CALL TERASE (IDINSTR,IDINSTR(19))                      ADDPNT   112
                  IF (NPAD.GT.0) CALL ADDARAY (XD,YD,NPAD,IDOCSTR,IDPT)     SOCT      35
120               CALL AERTRN                                               INHALE    49
                  END                                                       ADDPNT   116
```

315

```
      SUBROUTINE ADINST                                                 ADINST     2
C                                                                       ADINST     3
C         THIS ROUTINE DISPLAYS THE INSTRUCTIONS FOR SELECTING          ADINST     4
C         A MODE OF ADDING A POINT                                      ADINST     5
C                                                                       ADINST     6
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),        6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT    1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBTTT     1
     X     ISTR,IPUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      CALL ORIGIN (-5.5,-5.43)                                          ADINST     9
      CALL SQUARE (0.,-2.,3.5,2.,IDINSTR(1),IGN,-0)                     ADINST    10
      CALL ANNOTA (1.03,.1,12HINSTRUCTIONS,12,IDINSTR(2),IGN,-0,-0,-0)  ADINST    11
      CALL ANNOTA (.5,-.5,16HPICK THE TYPE OF,16,IDINSTR(3),IGN,-0,-0,  ADINST    12
     1     -0)                                                          ADINST    13
      CALL ANNOTA (.5,-.75,18HPOINT ENTRY METHOD,18,IDINSTR(4),         ADINST    14
     1     IGN,-0,-0,-0)                                                ADINST    15
      CALL ANNOTA (.5,-1.,16HYOU WISH TO USE.,16,IDINSTR(5),            ADINST    16
     1     IGN,-0,-0,-0)                                                ADINST    17
      IDINSTR(19)=5                                                     ADINST    18
      RETURN                                                            ADINST    19
      END                                                               ADINST    20
```

```
      SUBROUTINE ADINST2                                                ADINST2    2
C                                                                       ADINST2    3
C         THIS ROUTINE DISPLAYS THE ADDITIONAL MENU TO ACCEPT           ADINST2    4
C         OR REJECT THE NEW POINT                                       ADINST2    5
C                                                                       ADINST2    6
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/ 2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),         6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBTTT     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77    3
      CALL ORIGIN (-1.,-4.5)                                            ADINST2    9
      CALL ANNOTA (0.,0.,15HCONTINUE ADDING,15,IDINSTR(1),IBUT,1,-0,-0) ADINST2    10
      CALL ANNOTA (0.,-1.,13HRESTORE CURVE,13,IDINSTR(2),IBUT,2,-0,-0)  ADINST2    11
      CALL ANNOTA (0.,-2.,17HSELECT NEXT CURVE,17,IDINSTR(3),IBUT,      ADINST2    12
     1    3,-0,-30)                                                     ADINST2    13
      IDINSTR(10)=3                                                     ADINST2    14
      CALL ORIGIN (-5.,-3.)                                             ADINST2    15
      RETURN                                                            ADINST2    16
      END                                                               ADINST2    17
```

```
      SUBROUTINE ADMENU                                                  ADMENU      2
C                                                                        ADMENU      3
C         THIS ROUTINE DISPLAYS THE ADD-A-POINT MENU                     ADMENU      4
C                                                                        ADMENU      5
      COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNEND(10),IOSPEC(19),IOCF(41),  /DISPLAY/   2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),          6JULY77     2
     X               IOINSTR(19),IODATA(20,4,2),NCIP,IOBLOCK(20),NCIPM  1SEPT       1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CY8777      1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77     3
      CALL ORIGIN (.65,-5.9)                                             ADMENU      8
      CALL SQUARE (0.,-1.25,2.25,1.25,IOMENU(1),IGN,-0)                  ADMENU      9
      CALL ANNOTA (.15,-.25,14HKEYBOARD ENTRY,14,IOMENU(2),IBUT,40,      20SEPT      42
     1    -0,-0)                                                         20SEPT      43
      CALL ANNOTA (.15,-1.,14HTRACKING CROSS,14,IOMENU(3),IBUT,41,-0,-0) ADMENU      11
      IOMENU(13)=3                                                       ADMENU      12
      RETURN                                                             ADMENU      13
      END                                                                ADMENU      14
```

```
1                SUBROUTINE ADDKEY (IXAD,IYAD,XV,YV)                                ADDKEY      2
                 C                                                                  ADDKEY      3
                 C     THIS ROUTINE ACCEPTS ENTRY OF ADD-A-POINT COORDINATES VIA    ADDKEY      4
                 C     THE KEYBOARD LIGHT REGISTER                                  ADDKEY      5
5                C                                                                  ADDKEY      6
                      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                           /PARMSV/    2
                      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX               /AXSCALE/   2
                   10 XV=FLTIN(9HENTER X-,8,-400,-900)                             ADDKEY      9
                      YV=FLTIN(9HENTER Y-,8,-400,-900)                             ADDKEY     10
10               C                                                                 ADDKEY     11
                 C     CONVERT TO PLOT DGUS                                        ADDKEY     12
                      IF ((XV.GT.XMAX).OR.(XV.LT.XMIN).OR.(YV.GT.YMAX).OR.        ADDKEY     13
                     1    (YV.LT.YMIN)) GO TO 100                                  ADDKEY     14
                      IXAD=IFIX(((XV-XMIN)*DGUPIN)/XINCR+XORDGU)                   ADDKEY     15
15                    IYAD=IFIX(((YV-YMIN)*DGUPIN)/YINCR+YORDGU)                   ADDKEY     16
                      RETURN                                                       ADDKEY     17
                 C     POINT OUT OF BOUNDS -- PRINT MESSAGE                        ADDKEY     18
                 C                                                                 ADDKEY     19
                  100 CALL ERROR4 (-4.,8.,18HPOINT OUT OF RANGE,18,6HADDKEY,6HINFORM) ADDKEY   20
20                    GO TO 10                                                     ADDKEY     21
                      END                                                          ADDKEY     22
```

```
      SUBROUTINE ADDTKC (IXAD,IYAD,XV,YV)                                ADDTKC     2
C                                                                        ADDTKC     3
C     THIS ROUTINE ACCEPTS ENTRY OF ADD-A-POINT COORDINATES              ADDTKC     4
C     VIA THE TRACKING CROSS                                             ADDTKC     5
C                                                                        ADDTKC     6
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYB777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77    3
      COMMON/DISPLAY/IDSTOP,IDDAD(31),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),IDDATAD(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                 /PARMSV/   2
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/  2
C                                                                        ADDTKC    11
C     DISPLAY INSTRUCTIONS, TRACKING CROSS AND ADD MENU                  ADDTKC    12
C                                                                        ADDTKC    13
      CALL TKINST                                                        ADDTKC    14
      CALL ORIGIN (-5.,-3.)                                              ADDTKC    15
      CALL GITCON (NCON,0,-1000)                                         ADDTKC    16
      CALL ANNOTA (5.,-3.,3HADD,3,IDADD,IBUT,43,-0,-0)                   ADDTKC    17
C                                                                        ADDTKC    18
   20 CALL GIBUT (0,NCON,IDDT,IDDC)                                      ADDTKC    19
      IF (IDDC.EQ.254) CALL STOP                                         ADDTKC    20
      IF (IDDC.NE.43) GO TO 20                                           ADDTKC    21
      CALL GITCOF (NCON,IXAD,IYAD)                                       ADDTKC    22
      IF ((IXAD.GT.600).OR.(IXAD.LT.-1000)) GO TO 50                     ADDTKC    23
      IF ((IYAD.GT.600).OR.(IYAD.LT.-600)) GO TO 50                      ADDTKC    24
      XV=((IXAD-XORDGU)*XINCR)/DGUPIN+XMIN                               ADDTKC    25
      YV=((IYAD-YORDGU)*YINCR)/DGUPIN+YMIN                               ADDTKC    26
      CALL TERASE (IDINSTR,IDINSTR(19))                                  ADDTKC    27
      CALL GIERAS (IDADD)                                                ADDTKC    28
      CALL GITCON (NCON,1600,1600)                                       ADDTKC    29
      RETURN                                                             ADDTKC    30
C                                                                        ADDTKC    31
   50 CALL ERRORM (-4.,5.,18HPOINT OUT OF RANGE,18,6HADDTKC,6HINFORM)    ADDTKC    32
      GO TO 20                                                           ADDTKC    33
      END                                                                ADDTKC    34
```

```
      SUBROUTINE ADDARAY (XD,YD,NPAD,IDDCSTR,IOPT)                      20SEPT    44
C                                                                       ADDARAY    3
C     ADDARAY ADDS NEW POINTS TO THE EXISTING CURVES                    ADDARAY    4
C                                                                       ADDARAY    5
      DIMENSION XD(1),YD(1),IOPT(1)                                     20SEPT    45
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/   2
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,      /PLTBEAD/  2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/  3
     X               IPFMN,IPFMV                                        /PLTBEAD/  4
      COMMON/DISPLAY/IOSTOP,IODAD(80),NU,NUEND(19),IOSPEC(19),IOCF(41), /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),          6JULY77    2
     X               IOINSTR(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT      1
C                                                                       ADDARAY   12
C     LOAD CURVE POINTS INTO X-Y ARRAY                                  ADDARAY   13
C                                                                       ADDARAY   14
      ICRVPTR=IODATAB(IDDCSTR,4,1)                                      ADDARAY   15
      CALL DMGET (ICNPTS,ICRVPTR,NPOINTS)                               ADDARAY   16
      CALL DMGET (ICREC,ICRVPTR,IRECPTR)                                ADDARAY   17
      CALL DMGET (ICXPTR,ICRVPTR,IXPTR)                                 ADDARAY   18
      CALL DMGET (ICYPTR,ICRVPTR,IYPTR)                                 ADDARAY   19
      KOMPCOD=7000000000018 + SHIFT(NPOINTS,18)                         ADDARAY   20
      CALL DMGET (KOMPCOD,IRECPTR,NREC)                                 ADDARAY   21
      CALL DMGET (KOMPCOD, IXPTR,X)                                     ADDARAY   22
      CALL DMGET (KOMPCOD, IYPTR,Y)                                     ADDARAY
      CALL DMGET (  ICSYM,ICRVPTR,ICSYMT)                               ADDARAY
      CALL DMGET (ICINDEP,ICRVPTR,ICINDET)                              ADDARAY   32
C                                                                       ADDARAY   35
C     INSERT NEW POINTS INTO ARRAYS AND SORT                            ADDARAY   36
C                                                                       ADDARAY   37
      IF (ICINDET.EQ.0) ICINDET=1HX                                     ADDARAY   38
      DO 50 J=1,NPAD                                                    ADDARAY   39
      IF (NPOINTS.GE.248) GO TO 60                                      ADDARAY   40
      NPOINTS=NPOINTS+1                                                 ADDARAY   41
      X(NPOINTS)=XD(J)                                                  ADDARAY   42
      Y(NPOINTS)=YD(J)                                                  ADDARAY   43
      NREC(NPOINTS)=999                                                 ADDARAY   44
   50 CONTINUE                                                          ADDARAY   45
      GO TO 70                                                          ADDARAY   46
   60 CALL ERROR (-4.,8.,43HYOU HAVE REACHED THE MAXIMUM OF 248 POINTS. ADDARAY   47
     1     ,43,7HADDARAY,6HINFORM)                                      ADDARAY   48
   70 CALL SORTER (NPOINTS,ICINDET)                                     ADDARAY   49
C                                                                       ADDARAY   50
C     REBUILD CURVE BEAD                                                ADDARAY   51
C                                                                       ADDARAY   52
      CALL DMSET ( ICNPTS,ICRVPTR,NPOINTS)                              ADDARAY   59
      CALL DMGTBD (NPOINTS,IRECPTR)                                     20SEPT    46
      CALL DMGTBD (NPOINTS,IXPTR)                                       20SEPT    47
      CALL DMGTBD (NPOINTS,IYPTR)                                       20SEPT    48
      CALL DMSET (ICREC,ICRVPTR,IRECPTR)                                20SEPT    49
      CALL DMSET (ICXPTR,ICRVPTR,IXPTR)                                 20SEPT    50
      CALL DMSET (ICYPTR,ICRVPTR,IYPTR)                                 20SEPT    51
      KOMPCOD=7000000000018 + SHIFT(NPOINTS,18)                         20SEPT    52
      CALL DMSET (KOMPCOD,IRECPTR,NREC)                                 20SEPT    53
      CALL DMSET (KOMPCOD,IXPTR,X)                                      20SEPT    54
```

```
                CALL CMSET (KOMPCOD,IVPTR,Y)                                    20SEPT     55
                C                                                               20SEPT     56
60              C     ERASE OLD CURVE -- DISPLAY NEW                            20SEPT     57
                C                                                               20SEPT     58
                CALL TERASE (IOPT,NPAC)                                         20SEPT     59
                CALL TERASE (IODAD(ICDATAB(IDDCSTR,2,1)),ICDATAB(IDDCSTR,3,1))  20SEPT     60
                CALL DISCRVE (ICRVPTR,ISYMT,INITSS,NUMDISP,IDDCSTR)             20SEPT     61
65              CALL UCDATAB (IDDCSTR,1,6HACTIVE,INITSS,NUMDISP,ICRVPTR)        20SEPT     62
                RETURN                                                          ADDARAY    71
                END                                                             ADDARAY    72
```

```
      SUBROUTINE TKINST                                                 TKINST     2
C                                                                       TKINST     3
C         THIS ROUTINE DISPLAYS THE INSTRUCTIONS FOR USING              TKINST     4
C         THE TRACKING CROSS                                            TKINST     5
C                                                                       TKINST     6
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBTTT     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/DISPLAY/IDSTOP,IDDAD(30),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITL(4),IDCO(23),IDMENU(28),           6JULY77    2
     X               IDINSTR(19),IDDATA8(20,4,2),NCIP,IDBLOCK(28),NCIFN 1SEPT      1
      CALL ORIGIN (-5.5,-5.43)                                          TKINST     9
      CALL SQUARE (0.,-2.,3.5,2.,IDINSTR(1),IGN,-0)                     TKINST    10
      CALL ANNOTA (1.03..1,12HINSTRUCTIONS,12,IDINSTR(2),IGN,-0,-0,-0)  TKINST    11
      CALL ANNOTA (.3,-.5,23HMOVE THE TRACKING CROSS,23,IDINSTR(3),     TKINST    12
     1     IGN,-0,-0,-0)                                                TKINST    13
      CALL ANNOTA (.3,-.75,18HTO THE LOCATION OF,18,IDINSTR(4),         TKINST    14
     1     IGN,-0,-0,-0)                                                TKINST    15
      CALL ANNOTA (.3,-1.,14HTHE NEW POINT.,14,IDINSTR(5),IGN,-0,-0,-0) TKINST    16
      CALL ANNOTA (.3,-1.5,11HPICK "ADD".,11,IDINSTR(6),IGN,-0,-0,-0)   TKINST    17
      IDINSTR(19)=6                                                     TKINST    18
      RETURN                                                            TKINST    19
      END                                                               TKINST    20
```

```
      SUBROUTINE DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NPLOT)
C
C     THIS ROUTINE DISPLAYS THE CURVE(SET OF POINTS) INDICATED BY
C     "ICRVPTR". THE PRIMARY INPUTS TO THE ROUTINE ARE THE POINTER
C     TO THE CURVE BEAD AND THE PLOT SCALE DATA. PRIMARY OUTPUTS
C     FROM THE ROUTINE ARE:
C          ISYMBOL - THE SYMBOL USED TO DISTINGUISH THIS CURVE FROM
C                    OTHER CURVES ON THE PLOT.
C           INITSS - THE DISPLAY ADDRESSES FOR THE CURVE ARE STORED
C                    IN ARRAY IDDAD. IDDAD(INITSS) CONTAINS THE
C                    DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED
C                    TO EXHIBIT THE CURVE.(INITSS-INITIAL SUBSCRIPT)
C          NUMDISP - NUMDISP CONTAINS THE NUMBER OF DISPLAYS
C                    REQUIRED TO EXHIBIT THE CURVE.
C          NPLOT   - NPLOT CONTAINS THE NUMBER OF THE
C                    PLOT IN THE TABLE
C     INITSS AND NUMDISP ARE NEEDED TO ERASE THE CURVE. THEY WILL BE
C     STORED IN THE CURVE DISPLAY ACCESS TABLE.
C
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT
      COMMON/DISPLAY/IDSTOP,IDDAD(99),NN,NNEND(10),IDSPEC(19),IDCF(41),
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO
      COMMON/PARMSV/XOROGU,YOROGU,OGUPIN
      COMMON/POINTS/X(248),Y(248),NREC(248)
      COMMON /SYMTAB/ ISYMTAB(20)
      DIMENSION MESG(7)
C
C     RECORD THE INITIAL VALUE OF THE IDDAD SUBSCRIPT.
C
      INITSS=(NPLOT-1)*4+1
      NN=INITSS-1
C
C     GET THE NUMBER OF POINTS ON THE CURVE, THE CURVE SYMBOL
C     NUMBER, THE RECORD BEAD POINTER, THE X-COORDINATE BEAD
C     POINTER, AND THE Y-COORDINATE BEAD POINTER FROM THE
C     CURVE BEAD.
C
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)
      CALL DMGET(ICSYM,ICRVPTR,ISYMBOL)
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)
C
C     IF THE NUMBER OF POINTS EXCEEDS THE PROGRAMMED LIMIT(248),
C     DISPLAY THE FIRST 248 POINTS ONLY.
C
      IF(NPOINTS .LE. 248) GO TO 10
      NPOINTS=248
      CALL DMSET(ICNPTS,ICRVPTR,NPOINTS)
      CALL ERROFM(0.,7.,62HPOINTS LIMIT EXCEEDED. THE FIRST 248 POINTS W
     XILL BE DISPLAYED.,62,7HDISCRVE,6HINFORM)
```

```
      C         CONVERT THE CURVE SYMBOL NUMBER INTO A CURVE SYMBOL.                DISCRVE    53
      C                                                                             DISCRVE    54
   60    10 ISYMBOL = ISYMTA3(ISYMBOL)                                              DISCRVE    55
      C                                                                             DISCRVE    56
      C         MOVE THE CONTENTS OF THE RECORD NUMBER BEAD, THE X-COORDINATE       DISCRVE    57
      C         BEAD AND THE Y-COORDINATE BEAD INTO CORE. THE RECORD NUMBER         DISCRVE    58
      C         BEAD IS A STATUS BEAD FOR THE POINTS IN THE OTHER TWO BEADS.        DISCRVE    59
   65 C              ACTIVE - THE RECORD NUMBER IS POSITIVE OR SHADDED. THE         DISCRVE    60
      C                       RECORD NUMBER IS SHADDED IF AND ONLY IF THE           DISCRVE    61
      C                       POINT HAS BEEN ADDED DURING AN EDIT.                  DISCRVE    62
      C                 INACTIVE - THE RECORD NUMBER IS NEGATIVE. A POINT IS       DISCRVE    63
      C                       INACTIVE IF AND ONLY IF IT HAS BEEN DELETED           DISCRVE    64
   70 C                       DURING AN EDIT.                                       DISCRVE    65
      C                                                                             DISCRVE    66
            KOMPCOD = 700000000018 + NPOINTS * 8**6                                 DISCRVE    67
            CALL DMGET(KOMPCOD,IRECPTR,NREC)                                        DISCRVE    68
            CALL DMGET(KOMPCOD,IXPTR,X)                                             DISCRVE    69
   75       CALL DMGET(KOMPCOD,IYPTR,Y)                                            DISCRVE    70
      C                                                                             DISCRVE    71
      C         LOOP THROUGH THE POINTS. IF A POINT IS INACTIVE OR NOT ON THE       DISCRVE    72
      C         PLOT, SKIP IT. OTHERWISE, CONVERT THE COORDINATES TO DGUS AND       DISCRVE    73
      C         PACK THE DESCRIPTION OF THE POINT INTO THE DESCRIPTION BUFFER.      DISCRVE    74
   80 C                                                                             DISCRVE    75
      12    DO 20 I=1,NPOINTS                                                       9JULY76     2
            IF(NREC(I) .LT. 0) GO TO 20                                             DISCRVE    77
            IF ((X(I).LT.XMIN).OR.(X(I).GT.XMAX))GO TO 20                           30JULY76    1
            IF ((Y(I).LT.YMIN).OR.(Y(I).GT.YMAX))GO TO 20                           30JULY76    2
   85       IX=IFIX(((X(I)-XMIN)*DGUPIN)/XINCR+XORDGU-6)                            5OCT        1
            IY=IFIX(((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU - 11.5)                 DISCRVE    81
            CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                               DISCRVE    82
            CALL GUAN(ISYMBOL,1,IBUF,NBYTE,MBYTE)                                   DISCRVE    83
            IF(NBYTE .LT. 312) GO TO 20                                             DISCRVE    84
   90 C                                                                             DISCRVE    85
      C         THE DESCRIPTION BUFFER IS FULL. FLUSH IT.                           DISCRVE    86
      C                                                                             DISCRVE    87
            NN=NN+1                                                                 DISCRVE    88
            CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                              DISCRVE    89
   95    20 CONTINUE                                                              DISCRVE    90
      C                                                                             DISCRVE    91
      C         ALL POINTS HAVE BEEN CONSIDERED. IS THE DESCRIPTION                 DISCRVE    92
      C         BUFFER EMPTY? IF NOT, FLUSH IT.                                     DISCRVE    93
      C                                                                             DISCRVE    94
  100       IF(NBYTE .EQ. 0) GO TO 30                                               DISCRVE    95
            NN=NN+1                                                                 DISCRVE    96
            CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                              DISCRVE    97
      30    NUMDISP = NN - INITSS + 1                                               DISCRVE    98
            RETURN                                                                  DISCRVE    99
  105       END                                                                     DISCRVE   100
```

```
1                FUNCTION FLOISN(NC,DISNO)                                  FLOISN      2
                 C       CONVERTS THE NC CHARACTER DISPLAY-CODE NUMBER (DISNO) TO    FLOISN      3
                 C       A FLOATING BCD NUMBER(FLOISN)                      FLOISN      4
                  1      ENCODE(10,100,FMT)NC                               FLOISN      5
5                 100    FORMAT(2H(E,I2,3H.0))                              FLOISN      6
                         DECODE(10,FMT,DISNO)FLOISN                         FLOISN      7
                         END                                                FLOISN      8
```

```
      FUNCTION FLTIN(TEXT,NC,IX,IY)                                          FLTIN      2
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                       CYB777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                      6JUL77     3
C     WRITES A MESSAGE STARTING AT IX,IY FOLLOWED BY A 10 CHARACTER          FLTIN      4
C     LIGHT REGISTER.                                                        FLTIN      5
C     WHEN A NUMERIC VALUE(INTEGER OR OTHERWISE)IS ENTERED INTO              FLTIN      6
C     THE LIGHTREGISTER FOLLOWED BY A RETURN         ,IT                     FLTIN      7
C     IS RETURNED TO THE CALLING PROGRAM AS THE FLOATING POINT               FLTIN      8
C     NUMBER FLTIN.                                                          FLTIN      9
C     THE NC CHARACTER MESSAGE TEXT IS ERASED.                               FLTIN     10
      DIMENSION TEXT(1)                                                      FLTIN     11
      CALL GIEOM (NCON,IEOM,6,200)                                           CYB777    29
      CALL GUPSET(IX,IY,0028,IBUF,NBYTE,MBYTE)                               FLTIN     13
      CALL GUAN(TEXT,NC,IBUF,NBYTE,MBYTE)                                    FLTIN     14
      CALL GIDISP(NCON,IBUF,NBYTE,IODAD,1,101)                               FLTIN     15
      NBYTE=0                                                                FLTIN     16
      IH=IX+NC*24                                                            FLTIN     17
   91 CALL GIANS(NCON,10,IH,IY)                                              FLTIN     18
   10 CALL GIBUT(0,NCON,IDOT,IDOC)                                           FLTIN     19
      IF(IDOC.NE.200)GO TO 10                                                FLTIN     20
      NCH=10                                                                 FLTIN     21
      CALL GIANE(NCON,NCH,IBUF)                                              FLTIN     22
      IF(NCH.LT.1)GO TO 91                                                   FLTIN     24
      CALL GIERAS(IODAD)                                                     FLTIN     25
      FLTIN=FLOISN(NCH,IBUF)                                                 FLTIN     26
      END                                                                    FLTIN     27
```

```
      SUBROUTINE SORTER(NPOINTS,IV)                                     SORTER     2
C                                                                       SORTER     3
C         THIS ROUTINE SORTS THE COORDINATE ARRAYS AND THE STATUS ARRAY. SORTER     4
C         THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT        SORTER     5
C         VARIABLE.                                                     SORTER     6
C                                                                       SORTER     7
C         NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.  SORTER     8
C              IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE SORTER     9
C                       THREE ARRAYS (X,Y, AND NREC) ARE SORTED SO THAT THE SORTER    10
C                       X - COORDINATE ARRAY IS IN ASCENDING ORDER.      SORTER    11
C                       IF IV = 1HY, THEN THE THREE ARRAYS ARE SORTED SO SORTER    12
C                       THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER. SORTER 13
C                                                                       SORTER    14
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      DO 10 L=1,NPOINTS,L                                               SORTER    16
      M = 2 * L - 1                                                     SORTER    17
   10 CONTINUE                                                          SORTER    18
C                                                                       SORTER    19
C         START LOOP                                                    SORTER    20
C                                                                       SORTER    21
   20 M = M / 2                                                         SORTER    22
      IF(M .EQ. 0) GO TO 70                                             SORTER    23
      K = NPOINTS - M                                                   SORTER    24
      DO 60 J=1,K                                                       SORTER    25
      L = J                                                             SORTER    26
   30 IF(L .LT. 1) GO TO 60                                             SORTER    27
C                                                                       SORTER    28
C         USE THE SORT KEY.                                             SORTER    29
C                                                                       SORTER    30
      IF(IV .EQ. 1HY) GO TO 40                                          SORTER    31
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORTER    32
      GO TO 50                                                          SORTER    33
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORTER    34
C                                                                       SORTER    35
C         EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.               SORTER    36
C                                                                       SORTER    37
   50 TEMP = X(L)                                                       SORTER    38
      X(L) = X(L+M)                                                     SORTER    39
      X(L+M) = TEMP                                                     SORTER    40
C                                                                       SORTER    41
C         EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.               SORTER    42
C                                                                       SORTER    43
      TEMP = Y(L)                                                       SORTER    44
      Y(L) = Y(L+M)                                                     SORTER    45
      Y(L+M) = TEMP                                                     SORTER    46
C                                                                       SORTER    47
C         EXCHANGE POSITIONS IN THE STATUS ARRAY.                       SORTER    48
C                                                                       SORTER    49
      ITEMP = NREC(L)                                                   SORTER    50
      NREC(L) = NREC(L+M)                                               SORTER    51
      NREC(L+M) = ITEMP                                                 SORTER    52
      L=L-M                                                             8JUNE76   15
      GO TO 30                                                          SORTER    54
   60 CONTINUE                                                          SORTER    55
      GO TO 20                                                          SORTER    56
   70 RETURN                                                            SORTER    57
      END                                                               SORTER    58
```

```
      OVERLAY (5,3)                                                    INHALE    50
      PROGRAM DELCRVE                                                  INHALE    51
C                                                                      DELCRVE    3
C         THIS ROUTINE DELETES THE CURVES SELECTED BY THE USER.        DELCRVE    4
C                                                                      DELCRVE    5
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                   /AXSCALE/  2
      COMMON/DISPLAY/IDSTOP,IDDAD(50),NN,NNEND(10),IDSPEC(19),IDCF(41),/DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),        6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPP 1SEPT    1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                 CY6777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                6JULY77    3
      COMMON/EDITFX/IDDCSTR                                            INHALE    52
C                                                                      DELCRVE    9
C         1 - RETRIEVE THE IDENTIFICATION OF EACH OF THE SELECTED CURVES DELCRVE   10
C             AND ERASE THE CURVE AND ITS DESCRIPTION.                 DELCRVE   11
C         2 - IN THE ACCESS TABLE CHANGE THE STATUS OF EACH ERASED CURVE DELCRVE   12
C             TO INACTIVE.                                             DELCRVE   13
C                                                                      DELCRVE   14
      GO TO 20                                                         DELCRVE   15
   10 N = 1                                                            DELCRVE   16
      CALL GIFSID(NCON,N,IDOT,IDDCSTR)                                 DELCRVE   17
      IF(N .EQ. 0) GO TO 30                                            DELCRVE   18
   20 NROW = IDDCSTR                                                   DELCRVE   19
      CALL TERASE (IDDAD(ICDATAB(NROW,2,1)),ICDATAB(NROW,3,1))         25MAY76   11
      CALL GIERAS(IDCD(NROW))                                          DELCRVE   21
      ICDATAB(NROW,1,1) = 9HINACTIVE                                   14APR76   40
      GO TO 10                                                         DELCRVE   23
C                                                                      DELCRVE   24
C         DISPLAY THE FOLLOWING MENU AND WAIT FOR A BUTTON PICK.       DELCRVE   25
C                          34 - RESTORE PLOT                           DELCRVE   26
C                          33 - SELECT NEXT CURVE                      DELCRVE   27
C                         254 - STOP                                   14APR76   41
C                                                                      DELCRVE   29
   30 CALL LCMENU                                                      DELCRVE   30
      CALL ORIGIN(-5.,-3.)                                             DELCRVE   31
   40 CALL GIBUT(0,NCON,IDOT,IDDC)                                     DELCRVE   32
      IF(IDDC .EQ. 34) GO TO 50                                        DELCRVE   33
      IF(IDDC .EQ. 33) GO TO 60                                        DELCRVE   34
      IF(IDDC .EQ. 254) CALL STOP                                      14APR76   42
      GO TO 40                                                         DELCRVE   36
C                                                                      DELCRVE   37
C         RESTORE PLOT WAS SELECTED. ERASE THE                         DELCRVE   38
C         MENU AND REDISPLAY THE DELETED CURVES.                       DELCRVE   39
C                                                                      DELCRVE   40
   50 CALL TERASE(IDMENU,IDMENU(13))                                   DELCRVE   41
      CALL REPEN                                                       INHALE    53
C                                                                      DELCRVE   44
C         SELECT NEXT CURVE WAS SELECTED. ERASE THE MENU, DELETE THE   DELCRVE   45
C         UNNECESSARY LINES FROM THE CURVE DISPLAY ACCESS TABLE, AND   DELCRVE   46
C         FINALIZE THE CURVE DELETIONS. (THE CHOICE SELECT NEXT CURVE  DELCRVE   47
C         IS IMPLICIT CONFIRMATION THAT THE USER DOES NOT WANT TO      DELCRVE   48
C         CHANGE THE DELETIONS HE HAS MADE.)                           DELCRVE   49
C                                                                      DELCRVE   50
   60 CALL TERASE(IDMENU,IDMENU(13))                                   DELCRVE   51
      CALL FORWEL                                                      DELCRVE   52
      CALL DELCARS                                                     DELCRVE   53
      CALL REPEN                                                       INHALE    54
```

```
      END                                                    DELCRVE   55
```

```
      SUBROUTINE DCMENU                                                 DCMENU      2
C                                                                       DCMENU      3
C         THIS ROUTINE DISPLAYS A MENU WHICH IS                         DCMENU      4
C         USED IN THE CURVE DELETING ROUTINE.                           DCMENU      5
C                                                                       DCMENU      6
      COMMON/DISPLAY/IDSTOP,IDDAD(60),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),         6JULY77     2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBTTT      1
     X     ISTR,IBUT,TMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77     3
      CALL ORIGIN(.25,-5.75)                                            DCMENU      9
      CALL SQUARE(0.,-1.25,2.54,1.25,IDMENU(1),IGN,-0)                  DCMENU     10
      CALL ANNOTA(1.93,.1,4HMENU,4,IDMENU(2),IGN,-0,-0,-0)              DCMENU     11
      CALL ANNOTA(.25,-.5,12HRESTORE PLOT,12,IDMENU(3),IBUT,34,-0,-0)   DCMENU     12
      CALL ANNOTA(.25,-1.,17HSELECT NEXT CURVE,17,IDMENU(4),IBUT,33,    DCMENU     13
     X            -0,-0)                                                DCMENU     14
      IDMENU(13) = 4                                                    DCMENU     15
      RETURN                                                            DCMENU     16
      END                                                               DCMENU     17
```

```
      SUBROUTINE DELGAR8                                                DELGAR8    2
C                                                                       DELGAR8    3
C     THIS ROUTINE:                                                     DELGAR8    4
C          1 - DELETES UNNECESSARY ROWS FROM THE CURVE DISPLAY          DELGAR8    5
C              ACCESS TABLE. UNNECESSARY ROWS ARE ROWS WHICH            DELGAR8    6
C              CORRESPOND TO DELETED CURVES.                            DELGAR8    7
C          2 - MOVES THE CURVE DESCRIPTIONS SO THAT THEIR POSITIONS      DELGAR8    8
C              ON THE SCREEN CORRESPONDS TO THE ORDER IN THE CURVE       DELGAR8    9
C              DISPLAY ACCESS TABLE. THERE WILL BE NO GAPS BETWEEN       DELGAR8   10
C              THE CURVE DESCRIPTIONS.                                   DELGAR8   11
C                                                                       DELGAR8   12
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),ICDATA8(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
C                                                                       DELGAR8   15
C     LOOP THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE. IF A     DELGAR8   16
C     CURVE IS INACTIVE, INCREMENT THE INACTIVE CURVE COUNTER(NIC)      DELGAR8   17
C     BY ONE AND GO TO THE NEXT ROW.                                    DELGAR8   18
C                                                                       DELGAR8   19
      NIC = 0                                                           DELGAR8   20
      DO 30 I=1,NCIP                                                    DELGAR8   21
      IF(ICDATA8(I,1) .EQ. 6HACTIVE) GO TO 10                           DELGAR8   22
      NIC = NIC + 1                                                     DELGAR8   23
      GO TO 30                                                          DELGAR8   24
C                                                                       DELGAR8   25
C     THE CURVE IS ACTIVE. IF THE COUNT OF INACTIVE CURVES IS ZERO,     DELGAR8   26
C     THEN THIS ROW CANNOT BE MOVED UP. IF THE COUNT IS GREATER THAN    DELGAR8   27
C     ZERO, THEN MOVE THIS ROW UP INTO THE GAP CREATED BY THE           DELGAR8   28
C     DELETED CURVE.                                                    DELGAR8   29
C                                                                       DELGAR8   30
   10 IF(NIC .EQ. 0) GO TO 30                                           DELGAR8   31
      DO 20 J=1,4                                                       DELGAR8   32
      ICDATA8(I - NIC,J) = ICDATA8(I,J)                                 DELGAR8   33
   20 CONTINUE                                                          DELGAR8   34
C                                                                       DELGAR8   35
C     MOVE THE CURVE DESCRIPTION UP.(GIMOVE COORDINATES                 DELGAR8   36
C     ARE IN DGUS RATHER THAN INCHES.)                                  DELGAR8   37
C                                                                       DELGAR8   38
C                                                                       DELGAR8   39
      IDCD(I - NIC) = IDCD(I)                                           DELGAR8   40
      IY=1000-100*(I-NIC)                                               8JUNE76    1
      CALL GIMOVE (-0,IY,-0,IDCD(I-NIC),-0,I-NIC)                       8JUNE76    2
   30 CONTINUE                                                          DELGAR8   43
C                                                                       DELGAR8   44
C     REDUCE THE NUMBER OF ACTIVE CURVES IN THE PLOT.                   DELGAR8   45
C                                                                       DELGAR8   46
      NCIP = NCIP - NIC                                                 DELGAR8   47
      RETURN                                                            DELGAR8   48
      END                                                               DELGAR8   49
```

```
      SUBROUTINE FCRVDEL                                                FCRVDEL    2
C                                                                       FCRVDEL    3
C     THIS ROUTINE FINALIZES THE DELETION OF CURVES FROM A PLOT. IT     FCRVDEL    4
C     LOOPS THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE. IF      FCRVDEL    5
C     THE STATUS OF A CURVE IN THE CURVE DISPLAY ACCESS TABLE IS        FCRVDEL    6
C     INACTIVE, THEN ITS STATUS IN THE DATA MANAGER FILE IS CHANGED     FCRVDEL    7
C     FROM ACTIVE TO INACTIVE.                                          FCRVDEL    8
C                                                                       FCRVDEL    9
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
     X               IDINST(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT      1
      DO 10 I=1,NCIP                                                    FCRVDEL   12
      IF (ICDATAB(I,1,1).NE.8HINACTIVE) GO TO 10                        5OCT      48
      CALL DMSET (ICSTAT,ICDATAB(I,4,1),8HINACTIVE)                     5OCT      49
      IST=(I-1)*2+1                                                     5OCT      50
      CALL GIDRAS (IDCF(IST),IDCF(IST+1))                               5OCT      51
      IDCF (IST)=IDCF(IST+1)=0                                          5OCT      52
      IF (ICDATAB(I,4,2).NE.0) CALL DMRLBD (ICDATAB(I,4,2))             5OCT      53
      ICDATAB(I,4,2)=0                                                  5OCT      54
      CALL DMSET (ICFIT,ICDATAB(I,4,1),0)                               5OCT      55
   10 CONTINUE                                                          FCRVDEL   15
      RETURN                                                            FCRVDEL   16
      END                                                               FCRVDEL   17
```

```
      OVERLAY (F,2)                                                          INHALE     55
      PROGRAM DELPNT                                                         INHALE     56
C                                                                            DELPNT      3
C         THIS ROUTINE DELETES POINTS ON THE SELECTED CURVE.                 DELPNT      4
C                                                                            DELPNT      5
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                         /AXSCALE/   2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,       /CRVBEAD/   2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                      /CRVBEAD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(99),NN,NNEND(10),IDSPEC(19),IDCF(41),      /DISPLAY/   2
     X                IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),             6JULY77     2
     X                IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPN    1SEPT       1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                       CYB777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                      6JULY77     3
      COMMON/PARMSV/XORUGU,YORUGU,UGUPIN                                     /PARMSV/    2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/    2
      COMMON/EDITFX/IDDCSTR                                                  INHALE     57
      INTEGER RECORD(3)                                                      DELPNT     12
C                                                                            DELPNT     13
C         IDDCSTR INDIRECTLY IDENTIFIES THE CURVE FROM WHICH POINTS ARE      DELPNT     14
C         TO BE DELETED. IDDCSTR IS THE NUMBER OF A ROW IN THE CURVE         DELPNT     15
C         DISPLAY ACCESS TABLE. THAT ROW CORRESPONDS TO A CURVE ON THE       DELPNT     16
C         SCREEN.                                                            DELPNT     17
C                                                                            DELPNT     18
C         IF MORE THAN ONE CURVE WAS PICKED, THEN THE CURVE FROM WHICH       DELPNT     19
C         POINTS WILL BE DELETED IS THE LAST ONE. WHEN CONTROL PASSES TO     DELPNT     20
C         THIS PROGRAM, IDDCSTR IS THE IDDC OF THE FIRST CURVE PICKED.       DELPNT     21
C         RETRIEVE THE IDDC OF THE LAST CURVE PICKED.                        DELPNT     22
C                                                                            DELPNT     23
   10 NROW = IDDCSTR                                                         DELPNT     24
      N=1                                                                    DELPNT     25
      CALL GIFSID(NCON,N,IDDT,IDDCSTR)                                       DELPNT     26
      IF(N .EQ. 1) GO TO 10                                                  DELPNT     27
C                                                                            DELPNT     28
C         1 - ERASE ALL CURVES AND DESCRIPTIONS BUT THE SELECTED CURVE       DELPNT     29
C             AND DESCRIPTION.                                               DELPNT     30
C         2 - IN THE CURVE DISPLAY ACCESS TABLE CHANGE THE STATUS OF         DELPNT     31
C             EACH ERASED CURVE FROM ACTIVE TO INACTIVE.                     DELPNT     32
C         3 - DISPLAY THE DIAMETER MENU, THE INSTRUCTIONS, AND THE           DELPNT     33
C             TRACKING CROSS.                                                DELPNT     34
C         4 - WAIT FOR A BUTTON PICK FROM THE FOLLOWING MENU:                DELPNT     35
C                   28 - 0.5 INCHES                                          DELPNT     36
C                   29 - 1.0 INCHES                                          DELPNT     37
C                   30 - 2.0 INCHES                                          DELRAD      1
C                   31 - RETURN                                              50CT       36
C                  254 - STOP                                                14APR76    43
C                                                                            DELPNT     39
      DO 20 I=1,NCIP                                                         DELPNT     40
      IF (I.EQ.NROW) GO TO 15                                                27AUG       5
      IDDATAB(I,1,1) = 8HINACTIVE                                            14APR76    44
      CALL TERASE(IDDAD(IDDATAB(I,2)),IDDATAB(I,3))                          DELPNT     43
      CALL GIERAS(IDCD(I))                                                   DELPNT     44
   15 CALL TERASE (IDCF(IDDATAB(I,2,2)),IDDATAB(I,3,2))                      27AUG       6
      IDDATAB(I,1,2)=8HINACTIVE                                              27AUG       7
   20 CONTINUE                                                               DELPNT     45
   25 CALL DIATST                                                            5MAY76     18
      CALL DIAMNU                                                            5MAY76     19
      CALL ORIGIN(-5.,-3.)                                                   5MAY76     20
```

```
      CALL GITCON(NCON,0,-1000)                                          DELPNT     48
   30 CALL GIBUT(0,NCON,IDDT,IDDC)                                       DELPNT     49
   60    IF(IDDC .EQ. 28) GO TO 40                                       DELPNT     50
         IF(IDDC .EQ. 29) GO TO 50                                       DELPNT     51
         IF (IDDC.EQ.30) GO TO 52                                        DELRAD      2
         IF (IDDC.EQ.31) GO TO 200                                       50CT       37
         IF(IDDC .EQ. 254) CALL STOP                                     14APR76    45
   65    GO TO 30                                                        DELPNT     53
C                                                                        DELPNT     54
C        1 - CONVERT THE RADIUS TO DIGIGRAPHIC UNITS                     DELPNT     55
C        2 - RETRIEVE THE COORDINATES OF THE TRACKING CROSS AND CHECK    DELPNT     56
C            TO SEE THAT THE TRACKING CROSS WAS ON THE PLOT. IF NOT,     DELPNT     57
   70 C      DISPLAY AN ERROR MESSAGE AND REPEAT THE SEQUENCE. IF SO,    DELPNT     58
C            ERASE THE INSTRUCTIONS, THE MENU, AND THE TRACKING CROSS.   DELPNT     59
   40 R = 50.                                                            DELPNT     60
      GO TO 55                                                           14APR76    46
   50 R = 100.                                                           DELPNT     62
   75    GO TO 55                                                        DELRAD      3
   52 R=200.                                                             DELRAD      4
   55 CALL GITCOF(NCON,IX,IY)                                            14APR76    47
      IF((IX .LT. -1000) .OR. (IX .GT. 800)) GO TO 60                    DELPNT     64
      IF((IY .LT. -600) .OR. (IY .GT. 600)) GO TO 60                     DELPNT     65
   80    GO TO 70                                                        DELPNT     66
   60 CALL ERROFM(1.0,7.0,39HTHE TRACKING CROSS WAS NOT ON THE PLOT.,    5MAY76     21
     X            39,6HDELPNT,6HINFORM)                                 DELPNT     68
      CALL GICLR(NCON)                                                   DELPNT     69
      CALL GITCON(NCON,0,-1000)                                          DELPNT     70
   85    GO TO 30                                                        DELPNT     71
   70 CALL TERASE(ICINSTR,ICINSTR(19))                                   DELPNT     72
      CALL TERASE(ICMENU,ICMENU(13))                                     DELPNT     73
      CALL GITCON(NCON,1600,1600)                                        DELPNT     74
C                                                                        DELPNT     75
   90 C     IN PREPARATION FOR DISPLAYING THE COORDINATES OF POINTS ON THE DELPNT     76
C           SCREEN:                                                      DELPNT     77
C                1 - DISPLAY THE COORDINATES HEADING, AND                DELPNT     78
C                2 - MOVE THE CONTENTS OF THE RECORD READ, THE           DELPNT     79
C                    X-COORDINATE READ, AND THE Y-COORDINATE             DELPNT     80
   95 C             READ INTO CORE.                                     DELPNT     81
C                                                                        DELPNT     82
      CALL ORIGIN(-5.,-3.)                                               DELPNT     83
      N=1                                                                DELPNT     84
      CALL ANNOTA(-3.28,6.1,11HCOORDINATES,11,IDBLOCK(N),IGN,-0,-0,-0)   DELPNT     85
  100 ICRVPTR =%ICDATA3(NROW,4)                                          DELPNT     86
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                 DELPNT     87
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                  DELPNT     88
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                   DELPNT     89
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                   DELPNT     90
  105 KOMPCOD = 700000000010 + SHIFT(NPOINTS,18)                         14APR76    48
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                   DELPNT     92
      CALL DMGET(KOMPCOD,IXPTR,X)                                        DELPNT     93
      CALL DMGET(KOMPCOD,IYPTR,Y)                                        DELPNT     94
C                                                                        DELPNT     95
  110 C     LOOP THROUGH THE POINTS ON THE CURVE. IF A POINT IS INACTIVE DELPNT     96
C           (NREC < 0) OR NOT ON THE PLOT, SKIP IT.                      DELPNT     97
C                                                                        DELPNT     98
      DO 90 I=1,NPOINTS                                                  DELPNT     99
      IF(NREC(I) .LT. 0) GO TO 90                                        DELPNT    100
```

```
      IF((X(I) .LT. XMIN) .OR. (X(I) .GT. XMAX)) GO TO 90                DELPNT  101
      IF((Y(I) .LT. YMIN) .OR. (Y(I) .GT. YMAX)) GO TO 90                DELPNT  102
C                                                                        DELPNT  103
C         THE POINT IS ACTIVE AND ON THE PLOT. CONVERT ITS COORDINATES   DELPNT  104
C         TO DGUS AND CHECK TO SEE IF IT IS WITHIN THE PICKED TOLERANCE  DELPNT  105
C         OF THE TRACKING CROSS.                                         DELPNT  106
C                                                                        DELPNT  107
      XDGU =((X(I) - XMIN) * DGUPIN)/XINCR + XORDGU                      DELPNT  108
      YDGU =((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU                      DELPNT  109
      IF((XDGU - FLOAT(IX))**2 + (YDGU - FLOAT(IY))**2) .GT. R**2)       DELPNT  110
     X                                                  GO TO 90        DELPNT  111
C                                                                        DELPNT  112
C         THE POINT IS WITHIN THE TOLERANCE OF THE TRACKING CROSS. IF    DELPNT  113
C         THE LIMIT OF TWELVE COORDINATES ON THE SCREEN HAS NOT BEEN     DELPNT  114
C         MET, ENCODE THE COORDINATES AND DISPLAY THEM AS A STRING       DELPNT  115
C         PICKABLE ITEM. IF THE LIMIT HAS BEEN MET, DISPLAY AN           DELPNT  116
C         INFORMATIVE ERROR MESSAGE.                                     DELPNT  117
C                                                                        DELPNT  118
      IF(N .LT. 13) GO TO 80                                             DELPNT  119
      CALL ERROR(0.,7.,69HNO MORE THAN TWELVE POINTS CAN BE DISPLAYED F  DELPNT  120
     XOR DELETION AT ONE TIME.,69,6HDELPNT,6HINFORM)                     DELPNT  121
      IDBLOCK(20) = 13                                                   DELPNT  122
      GO TO 100                                                          DELPNT  123
   80 N = N + 1                                                          DELPNT  124
      ENCODE(27,1000,RECORD) X(I),Y(I)                                   DELPNT  125
 1000 FORMAT("(",G12.6,",",G12.6,")")                                    DELPNT  126
      YY = 6. - .5 * FLOAT(N - 1)                                        DELPNT  127
      CALL ANNOTA(-4.24,YY,RECORD,27,IDBLOCK(N),ISTR,I,-0,-0)            DELPNT  128
   90 CONTINUE                                                           DELPNT  129
      IF (N.GT.1) GO TO 95                                               27MAY76   3
      CALL ERROR(0.,7.,50HTHERE ARE NO POINTS WITHIN TRACKING CROSS DIA  27MAY76   4
     1METER,50,6HDELPNT,6HINFORM)                                        27MAY76   5
      CALL GIEPAS (IDBLOCK(1))                                           27MAY76   6
      GO TO 25                                                           27MAY76   7
   95 IDBLOCK(20)=N                                                      27MAY76   8
C                                                                        DELPNT  131
C         DISPLAY THE DELETE BUTTON AND INSTRUCTIONS FOR CHOOSING WHICH  DELPNT  132
C         POINTS ARE TO BE DELETED. WAIT FOR A BUTTON PICK FROM THE      DELPNT  133
C         FOLLOWING MENU:                                                DELPNT  134
C              30 - DELETE                                               DELPNT  135
C             254 - STOP                                                 14APR76  49
C                                                                        DELPNT  137
  100 CALL ANNOTA(-2.98,-1.,6HDELETE,6,IDMENU(1),IBUT,30,-0,-0)          DELPNT  138
      IDMENU(1) = 1                                                      DELPNT  139
      CALL DELINST                                                       DELPNT  140
      CALL ORIGIN(-5.,-3.)                                               DELPNT  141
  110 CALL GIBUT(0,NCON,IDDT,IDDC)                                       DELPNT  142
      IF(IDDC .EQ. 30) GO TO 115                                         DELPNT  143
      IF(IDDC .EQ. 254) CALL STOP                                        14APR76  50
      GO TO 110                                                          DELPNT  145
C                                                                        DELPNT  146
C         RETRIEVE THE IDENTIFICATION OF THE PICKED POINTS. IF NO POINTS DELPNT  147
C         WERE PICKED, PRINT AN INFORMATIVE ERROR MESSAGE. IF POINTS     DELPNT  148
C         WERE PICKED, CHANGE THEIR STATUS TO INACTIVE(NREC < 0).        DELPNT  149
C                                                                        DELPNT  150
  115 N = 1                                                              DELPNT  151
      CALL GIFSID(NCON,N,IDDT,IDDC)                                      DELPNT  152
```

```
                IF(N .EQ. 1) GO TO 130                                          DELPNT    153
                CALL ERROFM(0.,7.,42HYOU DID NOT PICK ANY POINTS TO BE DELETED.,DELPNT    154
               X            42,6HDELPNT,6HINFORM)                               DELPNT    155
      175       GO TO 110                                                       DELPNT    156
            120 CALL GIFSID(NCON,N,IDOT,IDOC)                                   DELPNT    157
                IF(N .EQ. 0) GO TO 140                                          DELPNT    158
            130 NREC(IDOC) = - NREC(IDOC)                                       DELPNT    159
                GO TO 120                                                       DELPNT    160
      180   C                                                                   DELPNT    161
            C       1 - MOVE THE CONTENTS OF THE STATUS ARRAY BACK INTO THE DATA DELPNT    162
            C           MANAGER FILE,                                           DELPNT    163
            C       2 - ERASE THE COORDINATES , THE INSTRUCTIONS, THE MENU,     14APR76    51
            C          AND THE CURVE, AND                                       14APR76    52
      185   C       3 - REDISPLAY THE CURVE WITH THE PICKED POINTS MISSING.      DELPNT    165
            C                                                                   DELPNT    166
            140 CALL DMSET(KCMPCOD,IRECPTR,NREC)                                DELPNT    167
                CALL TERASE(TCBLOCK,ICBLOCK(20))                                DELPNT    168
                CALL TERASE(ICINSTR,ICINSTR(19))                                14APR76    53
      190       CALL TERASE(ICMENU,ICMENU(13))                                  DELPNT    169
                CALL TERASE(ICDAD(ICDATAB(NROW,2)),ICDATAB(NROW,3))             DELPNT    170
                NN = 0                                                          DELPNT    171
                CALL DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NROW)              ADPTPRE    15
                CALL UCDATAB(NROW,1,6HACTIVE,INITSS,NUMDISP,ICRVPTR)            14APR76    54
      195   C                                                                   DELPNT    174
            C       DISPLAY THE FOLLOWING MENU AND WAIT FOR A BUTTON PICK FROM ITS DELPNT    175
            C                          31 - CONTINUE DELETING                   DELPNT    176
            C                          32 - RESTORE CURVE                       DELPNT    177
            C                          33 - SELECT NEXT CURVE                   DELPNT    178
      200   C                         254 - STOP                                14APR76    55
            C                                                                   DELPNT    180
                CALL CRSMENU                                                    DELPNT    181
                CALL ORIGIN(-5.,-3.)                                            5MAY76     22
            150 CALL GIBUT(0,NCON,IDOT,IDOC)                                    DELPNT    182
      205       IF(IDOC .EQ. 31) GO TO 160                                      DELPNT    183
                IF(IDOC .EQ. 32) GO TO 170                                      DELPNT    184
                IF(IDOC .EQ. 33) GO TO 190                                      DELPNT    185
                IF(IDOC .EQ. 254) CALL STOP                                     14APR76    56
                GO TO 150                                                       DELPNT    187
      210   C                                                                   DELPNT    188
            C       CONTINUE DELETING WAS SELECTED.                             DELPNT    189
            C                                                                   DELPNT    190
            160 CALL TERASE(ICMENU,ICMENU(13))                                  DELPNT    191
                GO TO 25 0                                                      DELPNT    192
      215   C                                                                   DELPNT    193
            C       RESTORE CURVE WAS SELECTED. ERASE THE EDITED                DELPNT    194
            C       CURVE AND REDISPLAY THE ORIGINAL ONE.                       DELPNT    195
            C                                                                   DELPNT    196
            170 CALL TERASE(ICDAD(ICDATAB(NROW,2)),ICDATAB(NROW,3))             DELPNT    197
      220       NN = 0                                                          DELPNT    198
                CALL DMGET(KCMPCOD,IRECPTR,NREC)                                DELPNT    199
                DO 180 I=1,NPOINTS                                              DELPNT    200
                IF(NREC(I) .LT. 0) NREC(I) = - NREC(I)                          DELPNT    201
            180 CONTINUE                                                        DELPNT    202
      225       CALL DMSET(KCMPCOD,IRECPTR,NREC)                                DELPNT    203
                CALL DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NROW)              ADPTPRE    16
                CALL UCDATAB(NROW,1,6HACTIVE,INITSS,NUMDISP,ICRVPTR)            5MAY76     23
                GO TO 150                                                       DELPNT    206
```

```
      C                                                                 DELPNT  207
230   C          SELECT NEXT CURVE WAS PICKED. ERASE                    DELPNT  208
      C          THE MENU AND REDISPLAY THE PLOT.                       DELPNT  209
      C                                                                 DELPNT  210
  190   CALL TERASE(ICMENU,IDMENU(13))                                  DELPNT  211
        CALL AERTRN                                                     INHALE   58
235   C                                                                 5OCT    38
      C          RETURN WITHOUT DELETING                                5OCT    39
  200   CALL TERASE (IDINSTR,IDINSTR(19))                               5OCT    40
        CALL TERASE (IDMENU,IDMENU(13))                                 5OCT    41
        CALL GITCON(NCON,1600,1600)                                     5OCT    42
240     CALL AERTRN                                                     INHALE   59
        END                                                             DELPNT  214
```

```
      SUBROUTINE CRSMENU                                                CRSMENU    2
C                                                                       CRSMENU    3
C       THIS ROUTINE DISPLAYS THE CONTINUE DELETING, RESTORE CURVE,     CRSMENU    4
C       SELECT NEXT CURVE MENU WHICH IS USED IN SUBROUTINE DELPNT.      CRSMENU    5
C                                                                       CRSMENU    6
      COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNENO(10),IOSPEC(19),IOCF(41), /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IOCD(23),IOMENU(20),        6JULY77    2
     X               IOINSTR(19),ICDATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT    1
      COMMON /GE/ IBUF(128),NCON,NBYTE,MBYTE,IGN,IBIN,                  CYBTTT     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77    3
      CALL ORIGIN(.25,-6.5)                                             CRSMENU    9
      CALL SQUARE(0.,-1.75,2.54,1.75,IOMENU(1),IGN,-0)                  CRSMENU   10
      CALL ANNOTA(1.03,.1,4HMENU,4,IOMENU(2),IGN,-0,-0,-0)              CRSMENU   11
      CALL ANNOTA(.25,-.5,17HCONTINUE DELETING,17,IOMENU(3),IBUT,31,    CRSMENU   12
     X            -0,-0)                                              CRSMENU   13
      CALL ANNOTA(.25,-1.,13HRESTORE CURVE,13,IOMENU(4),IBUT,32,-0,-0)  CRSMENU   14
      CALL ANNOTA(.25,-1.5,17HSELECT NEXT CURVE,17,IOMENU(5),IBUT,33,   CRSMENU   15
     X            -0,-0)                                              CRSMENU   16
      IOMENU(13) = 6                                                    CRSMENU   17
      RETURN                                                            CRSMENU   18
      END                                                               CRSMENU   19
```

```
      SUBROUTINE DELINST                                                DELINST    2
C                                                                       DELINST    3
C       THIS ROUTINE DISPLAYS THE INSTRUCTIONS FOR                      DELINST    4
C       DELETING POINTS WHOSE COORDINATES ARE DISPLAYED.                DELINST    5
C                                                                       DELINST    6
      COMMON/DISPLAY/IOSTOP,IDDAD(31),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/ 2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),         6JULY77    2
     X               IDINSTR(19),ICDATA8(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT     1
      COMMON /GP/ IBUF(129),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777     1
     X     ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      CALL ORIGIN(-6.,-5.5)                                             DELINST    9
      CALL SQUARE(0.,-1.5,4.1,1.5,IDINSTR(1),IGN,-0)                    DELINST   10
      CALL ANNOTA(1.2,.1,12HINSTRUCTIONS,12,IDINSTR(2),IGN,-0,-0,-0)    DELINST   11
      CALL ANNOTA(.25,-.5,29H1.STRING PICK THE COORDINATES,29,          DELINST   12
     X            IDINSTR(3),IGN,-),-0,-0)                            DELINST   13
      CALL ANNOTA(.49,-.75,28HOF THE POINTS TO BE DELETED.,28,          DELINST   14
     X            IDINSTR(4),IGN,-0,-0,-0)                            DELINST   15
      CALL ANNOTA(.25,-1.25,14H2.PICK DELETE.,14,IDINSTR(5),IGN,        DELINST   16
     X            -0,-0,-0)                                           DELINST   17
      IDINSTR(19) = 5                                                   DELINST   18
      RETURN                                                            DELINST   19
      END                                                               DELINST   20
```

```
      SUBROUTINE DIAINST                                                DIAINST    2
C                                                                       DIAINST    3
C     THIS ROUTINE DISPLAYS THE INSTRUCTIONS FOR USING                  DIAINST    4
C     THE TRACKING CROSS AND THE DIAMETER MENU.                         DIAINST    5
C                                                                       DIAINST    6
      COMMON/DISPLAY/IDSTOP,IDOAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIFM 1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777     1
     X    ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77    3
      CALL ORIGIN(-5.5,-5.43)                                           DIAINST    9
      CALL SQUARE(0.,-2.,3.5,2.,IDINSTR(1),IGN,-0)                      DIAINST   10
      CALL ANNOTA(1.83,.1,12HINSTRUCTIONS,12,IDINSTR(2),IGN,-0,-0,-0)   DIAINST   11
      CALL ANNOTA(.25,-.5,25H1.MOVE THE TRACKING CROSS,25,IDINSTR(3),   DIAINST   12
     X            IGN,-0,-0,-0)                                         DIAINST   13
      CALL ANNOTA(.49,-.75,18HTO THE CENTER OF A,18,IDINSTR(4),IGN,     DIAINST   14
     X            -0,-0,-0)                                             DIAINST   15
      CALL ANNOTA(.49,-1.,21HCIRCLE CONTAINING THE,21,IDINSTR(5),IGN,   DIAINST   16
     X            -0,-0,-0)                                             DIAINST   17
      CALL ANNOTA(.49,-1.25,21HPOINTS TO BE DELETED.,21,IDINSTR(6),     DIAINST   18
     X            IGN,-0,-0,-0)                                         DIAINST   19
      CALL ANNOTA(.25,-1.75,18H2.PICK A DIAMETER.,18,IDINSTR(7),IGN,    DIAINST   20
     X            -0,-0,-0)                                             DIAINST   21
      IDINSTR(19) = 7                                                   DIAINST   22
      RETURN                                                            DIAINST   23
      END                                                               DIAINST   24
```

```
      SUBROUTINE DIAMENU                                                  DIAMENU    2
C                                                                         DIAMENU    3
C         THIS ROUTINE DISPLAYS THE DIAMETER MENU.                        DIAMENU    4
C                                                                         DIAMENU    5
      COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNEND(10),IOSPEC(19),IOCF(41),   /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IOCD(23),IOMENU(20),          6JULY77    2
     X               IOINSTR(19),IODATA3(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                    CYBER77    1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77    3
      CALL ORIGIN(.65,-5.8)                                               DIAMENU    8
      CALL SQUARE(0.,-2.,1.7,2.,IOMENU(1),IGN,-0,-0,-0)                   50CT      45
      CALL ANNOTA(.24,.1,8HDIAMETER,8,IOMENU(2),IGN,-0,-0,-0)             DIAMENU   10
      CALL ANNOTA(.25,-.5,10H0.5 INCHES,10,IOMENU(3),IBUT,28,-0,-0)       DIAMENU   11
      CALL ANNOTA(.25,-1.,10H1.0 INCHES,10,IOMENU(4),IBUT,29,-0,-0)       DIAMENU   12
      CALL ANNOTA (.25,-1.5,10H2.0 INCHES,10,IOMENU(5),IBUT,30,-0,-0)     DELRAD     6
      CALL ANNOTA (.27,-1.95,6HRETURN,6,IOMENU(6),IBUT,31,-0,-0)          50CT      46
      IOMENU(13)=6                                                        50CT      47
      RETURN                                                              DIAMENU   14
      END                                                                 DIAMENU   15
```

```
      SUBROUTINE DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NPLOT)          ADPTPRE     1
C                                                                        DISCRVE     3
C     THIS ROUTINE DISPLAYS THE CURVE(SET OF POINTS) INDICATED BY        DISCRVE     4
C     "ICRVPTR". THE PRIMARY INPUTS TO THE ROUTINE ARE THE POINTER       DISCRVE     5
C     TO THE CURVE BEAD AND THE PLOT SCALE DATA. PRIMARY OUTPUTS         DISCRVE     6
C     FROM THE ROUTINE ARE:                                              DISCRVE     7
C          ISYMBOL - THE SYMBOL USED TO DISTINGUISH THIS CURVE FROM      DISCRVE     8
C                    OTHER CURVES ON THE PLOT.                           DISCRVE     9
C           INITSS - THE DISPLAY ADDRESSES FOR THE CURVE ARE STORED      DISCRVE    10
C                    IN ARRAY IDDAD. IDDAD(INITSS) CONTAINS THE          DISCRVE    11
C                    DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED       DISCRVE    12
C                    TO EXHIBIT THE CURVE.(INITSS-INITIAL SUBSCRIPT)     DISCRVE    13
C          NUMDISP - NUMDISP CONTAINS THE NUMBER OF DISPLAYS             DISCRVE    14
C                    REQUIRED TO EXHIBIT THE CURVE.                      DISCRVE    15
C          NPLOT   - NPLOT CONTAINS THE NUMBER OF THE                    ADPTPRE     2
C                    PLOT IN THE TABLE                                   ADPTPRE     3
C     INITSS AND NUMDISP ARE NEEDED TO ERASE THE CURVE. THEY WILL BE     DISCRVE    16
C     STORED IN THE CURVE DISPLAY ACCESS TABLE.                          DISCRVE    17
C                                                                        DISCRVE    18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/   2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,   /CRVBEAD/   2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                  /CRVBEAD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(99),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),           6JULY77     2
     X               IDINSTP(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT       1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYBTTT      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77     3
      COMMON/PARMSW/XORDGU,YORDGU,DGUPIN                                 /PARMSW/    2
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/    2
      COMMON /SYMTAB/ ISYMTAB(20)                                        /SYMTAB/    2
      DIMENSION MESG(7)                                                  9JULY76     1
C                                                                        DISCRVE    28
C     RECORD THE INITIAL VALUE OF THE IDDAD SUBSCRIPT.                   DISCRVE    29
C                                                                        DISCRVE    30
      INITSS=(NPLOT-1)*4+1                                               ADPTPRE     4
      NN=INITSS-1                                                        ADPTPRE     5
C                                                                        DISCRVE    32
C     GET THE NUMBER OF POINTS ON THE CURVE, THE CURVE SYMBOL            DISCRVE    33
C     NUMBER, THE RECORD BEAD POINTER, THE X-COORDINATE BEAD             DISCRVE    34
C     POINTER, AND THE Y-COORDINATE BEAD POINTER FROM THE                DISCRVE    35
C     CURVE BEAD.                                                        DISCRVE    36
C                                                                        DISCRVE    37
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                 DISCRVE    38
      CALL DMGET(ICSYM,ICRVPTR,ISYMBOL)                                  DISCRVE    39
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                  DISCRVE    40
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                   DISCRVE    41
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                   DISCRVE    42
C                                                                        DISCRVE    43
C     IF THE NUMBER OF POINTS EXCEEDS THE PROGRAMMED LIMIT(248),         DISCRVE    44
C     DISPLAY THE FIRST 248 POINTS ONLY.                                 DISCRVE    45
C                                                                        DISCRVE    46
      IF(NPOINTS .LE. 248) GO TO 10                                      DISCRVE    47
      NPOINTS=248                                                        DISCRVE    48
      CALL DMSET(ICNPTS,ICRVPTR,NPOINTS)                                 DISCRVE    49
      CALL ERROR(6,7,62HPOINTS LIMIT EXCEEDED. THE FIRST 248 POINTS W    DISCRVE    50
     XILL BE DISPLAYED.,62,7HDISCRVE,6HINFORM)                           DISCRVE    51
C                                                                        DISCRVE    52
```

```
                C         CONVERT THE CURVE SYMBOL NUMBER INTO A CURVE SYMBOL.                DISCRVE    53
                C                                                                             DISCRVE    54
     60           10 ISYMBOL = ISYMTAB(ISYMBOL)                                               DISCRVE    55
                C                                                                             DISCRVE    56
                C         MOVE THE CONTENTS OF THE RECORD NUMBER BEAD, THE X-COORDINATE       DISCRVE    57
                C         BEAD AND THE Y-COORDINATE BEAD INTO CORE. THE RECORD NUMBER         DISCRVE    58
                C         BEAD IS A STATUS BEAD FOR THE POINTS IN THE OTHER TWO BEADS.        DISCRVE    59
     65         C              ACTIVE - THE RECORD NUMBER IS POSITIVE OR SHADDED. THE         DISCRVE    60
                C                       RECORD NUMBER IS SHADDED IF AND ONLY IF THE          DISCRVE    61
                C                       POINT HAS BEEN ADDED DURING AN EDIT.                 DISCRVE    62
                C            INACTIVE - THE RECORD NUMBER IS NEGATIVE. A POINT IS           DISCRVE    63
                C                       INACTIVE IF AND ONLY IF IT HAS BEEN DELETED          DISCRVE    64
     70         C                       DURING AN EDIT.                                      DISCRVE    65
                C                                                                             DISCRVE    66
                     KOMPCOD = 700000000018 + NPOINTS * 8**6                                  DISCRVE    67
                     CALL DMGET(KOMPCOD,IRECPTR,NREC)                                         DISCRVE    68
                     CALL DMGET(KOMPCOD,IXPTR,X)                                              DISCRVE    69
     75              CALL DMGET(KOMPCOD,IYPTR,Y)                                              DISCRVE    70
                C                                                                             DISCRVE    71
                C         LOOP THROUGH THE POINTS. IF A POINT IS INACTIVE OR NOT ON THE       DISCRVE    72
                C         PLOT, SKIP IT. OTHERWISE, CONVERT THE COORDINATES TO DGUS AND       DISCRVE    73
                C         PACK THE DESCRIPTION OF THE POINT INTO THE DESCRIPTION BUFFER.      DISCRVE    74
     80         C                                                                             DISCRVE    75
                     12 DO 20 I=1,NPOINTS                                                     9JULY76     2
                        IF(NREC(I) .LT. 0) GO TO 20                                           DISCRVE    77
                        IF ((X(I).LT.XMIN).OR.(X(I).GT.XMAX))GO TO 20                         30JULY76    1
                        IF ((Y(I).LT.YMIN).OR.(Y(I).GT.YMAX))GO TO 20                         30JULY76    2
     85                 IX=IFIX(((X(I)-XMIN)*DGUPIN)/XINCR+XORDGU-6)                          5OCT        1
                        IY=IFIX(((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU - 11.5)               DISCRVE    81
                        CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                             DISCRVE    82
                        CALL GUAN(ISYMBOL,1,IBUF,NBYTE,MBYTE)                                 DISCRVE    83
                        IF(NBYTE .LT. 312) GO TO 20                                           DISCRVE    84
     90         C                                                                             DISCRVE    85
                C         THE DESCRIPTION BUFFER IS FULL. FLUSH IT.                           DISCRVE    86
                C                                                                             DISCRVE    87
                        NN=NN+1                                                               DISCRVE    88
                        CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                            DISCRVE    89
     95              20 CONTINUE                                                              DISCRVE    90
                C                                                                             DISCRVE    91
                C         ALL POINTS HAVE BEEN CONSIDERED. IS THE DESCRIPTION                 DISCRVE    92
                C         BUFFER EMPTY? IF NOT, FLUSH IT.                                     DISCRVE    93
                C                                                                             DISCRVE    94
    100                 IF(NBYTE .EQ. 0) GO TO 30                                             DISCRVE    95
                        NN=NN+1                                                               DISCRVE    96
                        CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                            DISCRVE    97
                     30 NUMDISP = NN - INITSS + 1                                             DISCRVE    98
                        RETURN                                                                DISCRVE    99
    105                 END                                                                   DISCRVE   100
```

```
1                OVERLAY (5,5)                                          REPLOTX    2
                 PROGRAM REPLOTX                                        REPLOTX    3
           C                                                            REPLOTX    4
           C THIS TS A DUMMY TO CALL THE PLOT ROUTINES                  REPLOTX    5
           C                                                            REPLOTX    6
5                CALL FEPLOT                                            REPLOTX    7
                 CALL AERTRN                                            REPLOTX    8
                 END                                                    REPLOTX    9
```

```
      SUBROUTINE DCRVDES(ICRVPTR,ISYMBOL,NCURVE)                                DCRVDES    2
C                                                                               DCRVDES    3
C         THIS ROUTINE BUILDS A CURVE DESCRIPTION AND DISPLAYS IT.              DCRVDES    4
C              ICRVPTR - THE CURVE BEAD POINTER.                                DCRVDES    5
C              ISYMBOL - THE CURVE PLOT SYMBOL.                                 DCRVDES    6
C               NCURVE - THE NUMBER OF THE CURVE. THE CURVES ARE                DCRVDES    7
C                        NUMBERED CHRONOLOGICALLY AS THEY APPEAR                DCRVDES    8
C                        ON THE SCREEN. THAT NUMBER IS USED IN                  DCRVDES    9
C                        THIS SUBROUTINE TO DETERMINE THE POSITION              DCRVDES   10
C                        ON THE SCREEN WHERE THE CURVE DESCRIPTION              DCRVDES   11
C                        WILL BE DISPLAYED.                                     DCRVDES   12
C                                                                               DCRVDES   13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,          /CRVBEAD/   2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                         /CRVBEAD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(99),NN,NNEND(10),IDSPEC(19),IDCF(41),         /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),                  6JULY77     2
     X               IDINSTR(19),ICDATA3(20,4,2),NCIP,IDBLOCK(20),NCIFM         1SEPT       1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                          CYBER77     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                         6JULY77     3
      INTEGER RECORD(3)                                                         DCRVDES   17
C                                                                               DCRVDES   18
C         GET THE PARAMETER NAME AND VALUE FROM THE CURVE BEAD, ENCODE          DCRVDES   19
C         THE CURVE DESCRIPTION, AND DISPLAY IT.                                DCRVDES   20
C                                                                               DCRVDES   21
      CALL DMGET(ICPARAN,ICRVPTR,NAME)                                          DCRVDES   22
      CALL DMGET(ICPARAV,ICRVPTR,VALUE)                                         DCRVDES   23
      IDEC=8                                                                    FIXINM     2
      DO 10 J=1,9                                                               FIXINM     3
      IF (VALUE/10.**(J-1).LE.1.) GO TO 20                                      FIXINM     4
   10 IDEC=IDEC-1                                                               FIXINM     5
      IDEC=9                                                                    FIXINM     6
   20 ENCODE (25,1000,RECORD) ISYMBOL,NAME,IDEC,VALUE                           FIXINM     7
 1000 FORMAT(A1,1X,A10,1X,F12.=)                                                FIXINM     8
      Y = 8.0 - .5 * FLOAT(NCURVE)                                              DCRVDES   26
      CALL ANNOTA (10.43,Y,RECORD,25,IDCD(NCURVE),IGN,NCURVE,-0,-0)             6JULY77     6
      RETURN                                                                    DCRVDES   28
      END                                                                       DCRVDES   29
```

```
      SUBROUTINE OCRVFIT(ITYPE,IV,X,Y,NPOINTS,C,NFP,TMIN,TMAX,TINCR,     OCRVFIT     2
     X          INITSS,NUMDISP,ICURVE)                                   28MAY76     1
C                                                                       OCRVFIT     4
C     THIS ROUTINE DISPLAYS THE CURVE FIT DEFINED BY PARAMETERS ONE     OCRVFIT     5
C     THROUGH SEVEN OVER THE INTERVAL DEFINED BY PARAMETERS EIGHT       OCRVFIT     6
C     AND NINE.                                                         OCRVFIT     7
C               SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND             13SEPT     18
C               6HAITKEN FOR A STRAIGHT FIT.                            13SEPT     19
C               SQUARES FIT AND 6HSPLINE FOR A SPLINE FIT.              OCRVFIT     9
C          IV - I - THE INDEPENDENT VARIABLE; EITHER 1HX OR 1HY.        OCRVFIT    10
C           X - I - THE ARRAY OF X-COORDINATES OF THE POINTS FOR        OCRVFIT    11
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      OCRVFIT    12
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     OCRVFIT    13
C           Y - I - THE ARRAY OF Y-COORDINATES OF THE POINTS FOR        OCRVFIT    14
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      OCRVFIT    15
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     OCRVFIT    16
C     NPOINTS - I - THE NUMBER OF POINTS WHOSE COORDINATES              OCRVFIT    17
C               ARE IN ARRAYS "X" AND "Y".                              OCRVFIT    18
C           C - I - THE ARRAY OF CURVE FIT PARAMETERS; POLYNOMIAL       OCRVFIT    19
C               COEFFICIENTS FOR A LEAST SQUARES FIT, DERIVATIVES       OCRVFIT    20
C               AT EACH POINT FOR A SPLINE FIT.                         OCRVFIT    21
C         NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF         OCRVFIT    22
C               POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT,        OCRVFIT    23
C               THE NUMBER OF POINTS FOR A SPLINE FIT.                  OCRVFIT    24
C        TMIN - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      OCRVFIT    25
C               THE DISPLAY OF THE CURVE FIT IS TO START.               OCRVFIT    26
C        TMAX - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      OCRVFIT    27
C               THE DISPLAY OF THE CURVE FIT IS TO STOP. (TMIN,TMAX)    OCRVFIT    28
C               IS THE INTERVAL OVER WHICH THE CURVE FIT IS TO BE       OCRVFIT    29
C               DISPLAYED.                                              OCRVFIT    30
C        TINCR - I - THE INCREMENT OF THE INDEPENDENT VARIABLE PER      OCRVFIT    31
C               TWENTIETH OF AN INCH. THE CURVE FIT WILL BE             OCRVFIT    32
C               APPROXIMATED BY TWENTY STRAIGHT LINES PER INCH.         OCRVFIT    33
C       INITSS - 0 - THE DISPLAY ADDRESSES OF THE CURVE FIT ARE         OCRVFIT    34
C               STORED IN ARRAY "IDCF". IDCF(INITSS) CONTAINS THE       OCRVFIT    35
C               DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED TO        OCRVFIT    36
C               EXHIBIT THE CURVE FIT.                                  OCRVFIT    37
C     NUMDISP - 0 - NUMDISP CONTAINS THE NUMBER OF DISPLAYS             OCRVFIT    38
C               REQUIRED TO EXHIBIT THE CURVE FIT.                      OCRVFIT    39
C                                                                       OCRVFIT    40
C     A 64 WORD DESCRIPTION BUFFER WILL HOLD THE DESCRIPTION BYTES      OCRVFIT    41
C     FOR 101 LINE SEGMENTS.                                            OCRVFIT    42
C          NUMBER     TYPE      BYTES                                   OCRVFIT    43
C            OF        OF       PER        TOTAL                       OCRVFIT    44
C           CALLS     CALL      CALL       BYTES                       OCRVFIT    45
C          *******   ******   *******   *******                        OCRVFIT    46
C             1       GDRSET      3          3                         OCRVFIT    47
C             1       GDSEGI      4          4                         OCRVFIT    48
C           101       GDSEG       3        303                         OCRVFIT    49
C             1       GDDISP     10         10                         OCRVFIT    50
C                                          320---TOTAL                 OCRVFIT    51
C     SINCE ANY CURVE FIT CAN BE DISPLAYED WITH AT MOST 100 LINE        OCRVFIT    52
C     SEGMENTS(NINE INCHES), EACH CURVE FIT CAN BE DISPLAYED WITH       OCRVFIT    53
C     AT MOST TWO BUFFERS.                                              OCRVFIT    54
C                                                                       OCRVFIT    55
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/   2
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NU,NWEND(10),IDSPEC(19),IDCF(41), /DISPLAY/   2
```

```
     X              IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),                6JULY77    2
     X              IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM       1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                        CYBTTT     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                       6JULY77    3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                      /PARMSV/   2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,        /CRVBEAD/  2
     X              ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                        /CRVBEAD/  3
      DIMENSION TEMP(14)                                                      13SEPT     20
      DIMENSION X(NPOINTS),Y(NPOINTS),C(NFP)                                  DCRVFIT    60
C                                                                             DCRVFIT    61
C         RECORD THE INITIAL VALUE OF THE "IDCF" SUBSCRIPT.                   DCRVFIT    62
C                                                                             DCRVFIT    63
      N=(ICURVE-1)*2                                                          28MAY76    2
      INITSS = N + 1                                                          DCRVFIT    65
C                                                                             DCRVFIT    66
      IF(IV .EQ. 1HY) GO TO 50                                                DCRVFIT    67
      IF(IV .EQ. 1HX) CONTINUE                                                DCRVFIT    68
C                                                                             DCRVFIT    69
C         INITIALIZE THE DESCRIPTION BUFFER TO THE FIRST POINT ON THE         DCRVFIT    70
C         CURVE FIT. THE Y-COORDINATE OF THE POINT DEPENDS UPON THE           DCRVFIT    71
C         TYPE OF CURVE FIT.                                                  DCRVFIT    72
C                                                                             DCRVFIT    73
      XX = XMIN                                                               DCRVFIT    74
      XXINCR = XINCR                                                          DCRVFIT    75
      XXMAX = XMAX                                                            DCRVFIT    76
    2 IF (ITYPE.EQ.1CHLEASTSQUAR) YY=P(XX,C,NFP)                              29APR      2
      IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)                     DCRVFIT    78
      IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,                20SEPT     [illegible]
     1    YY,TEMP,IER)                                                        13SEPT     22
      IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX)            29APR      3
     1     GO TO 5                                                          27MAY      3
      XX=XX+XXINCR                                                            29APR      5
      IF (XX.GT.XXMAX) GO TO 32                                               29APR      6
      GO TO 2                                                                 29APR      7
    5 IF (IER.EQ.2) GO TO 100                                                 29APR      8
      IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)                   DCRVFIT    79
      IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)                   DCRVFIT    80
      NBYTE = 0                                                               DCRVFIT    81
   10 CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                               DCRVFIT    82
      CALL GUSEGI(IX,IY,ISTYLE,IBUF,NBYTE,MBYTE)                              DCRVFIT    83
C                                                                             DCRVFIT    84
C         INCREMENT THROUGH THE INTERVAL AND PACK THE DESCRIPTION BYTES       DCRVFIT    85
C         OF APPROXIMATING LINE SEGMENTS INTO THE DESCRIPTION BUFFER.         DCRVFIT    86
C                                                                             DCRVFIT    87
   20 XX = XX + XXINCR                                                        DCRVFIT    88
      IF(ITYPE .EQ. 1CHLEASTSQUAR) YY = P(XX,C,NFP)                           DCRVFIT    89
      IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)                     DCRVFIT    90
      IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,                20SEPT     77
     1    YY,TEMP,IER)                                                        13SEPT     24
      IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 30                                  14MARCH    17
      IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 30                                  14MARCH    18
      IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)                   DCRVFIT    91
      IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)                   DCRVFIT    92
      CALL GUSEG(IX,IY,IBEAM)                                                 DCRVFIT    93
C                                                                             DCRVFIT    94
C         CHECK TO SEE IF THE DESCRIPTION BUFFER IS FULL.                     DCRVFIT    95
C         IF SO, FLUSH IT. IF NOT, CONTINUE TO FILL IT.                       DCRVFIT    96
```

```
      C                                                                      14APR76   71
            IF(NBYTE .LT. 312) GO TO 30                                      DCRVFIT   97
            N = N + 1                                                        DCRVFIT   98
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                         DCRVFIT   99
            IF(XX .LT. XXMAX) GO TO 10                                       DCRVFIT  100
      C                                                                      DCRVFIT  101
      C        THE DESCRIPTION BUFFER IS NOT FULL. CHECK TO SEE IF THE CURVE DCRVFIT  102
      C         FIT IS COMPLETE. IF NOT, CONTINUE TO GENERATE THE DESCRIPTION DCRVFIT  103
      C         BYTES FOR THE CURVE FIT. IF SO, DISPLAY THE CURVE FIT.       DCRVFIT  104
      C                                                                      DCRVFIT  105
         30 IF(XX .LT. XXMAX) GO TO 20                                       DCRVFIT  106
         32 IF (NBYTE.EQ.0) GO TO 40                                         29APR     9
            N = N + 1                                                        DCRVFIT  108
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                         DCRVFIT  109
         40 NUMDISP = N - INITSS + 1                                         DCRVFIT  110
            IDCF(41)=N                                                       25MAY76   15
            RETURN                                                           DCRVFIT  111
      C                                                                      DCRVFIT  112
      C        "Y" IS THE INDEPENDENT VARIABLE. COMMENTS                     DCRVFIT  113
      C        ANALOGOUS TO THOSE ABOVE APPLY.                               DCRVFIT  114
      C                                                                      DCRVFIT  115
         50 YY = YMIN                                                        DCRVFIT  116
            YYINCR = YINCR                                                   DCRVFIT  117
            YYMAX = YMAX                                                     DCRVFIT  118
         52 IF (ITYPE.EQ.10HLEASTSQUAR) XX=P(YY,C,NFP)                       29APR    10
            IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)              DCRVFIT  120
            IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,         20SEPT   78
           1    XX,TEMP,IER)                                                13SEPT   26
            IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX)     29APR    11
           1       GO TO 55                                                  29APR    12
            YY=YY+YYINCR                                                     29APR    13
            IF (YY.GT.YYMAX) GO TO 82                                        29APR    14
            GO TO 52                                                         29APR    15
         55 IF (IER.EQ.2) GO TO 100                                          29APR    16
            IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)            DCRVFIT  121
            IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)            DCRVFIT  122
            NBYTE = 0                                                        DCRVFIT  123
         60 CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                        DCRVFIT  124
            CALL GUSEGI(IX,IY,ISTYLE,IBUF,NBYTE,MBYTE)                       DCRVFIT  125
         70 YY = YY + YYINCR                                                 DCRVFIT  126
            IF(ITYPE .EQ. 10HLEASTSQUAR) XX = P(YY,C,NFP)                    DCRVFIT  127
            IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)              DCRVFIT  128
            IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,         20SEPT   79
           1    XX,TEMP,IER)                                                13SEPT   28
            IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 80                           14MARCH  21
            IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 80                           14MARCH  22
            IF (IER.EQ.2) GO TO 100                                          3MARCH   19
            IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)            DCRVFIT  129
            IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)            DCRVFIT  130
            CALL GUSEG(IX,IY,IBEAM)                                          DCRVFIT  131
            IF(NBYTE .LT. 312) GO TO 80                                      DCRVFIT  132
            N = N + 1                                                        DCRVFIT  133
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                         DCRVFIT  134
            IF(YY .LT. YYMAX) GO TO 60                                       DCRVFIT  135
         80 IF(YY .LT. YYMAX) GO TO 70                                       DCRVFIT  136
         82 IF (NBYTE.EQ.0) GO TO 90                                         29APR    17
            N = N + 1                                                        DCRVFIT  138
```

```
                  CALL GIDISP(NCON,I8UF,N3YTE,IOCF(N),IGN)                          OCRVFIT   139
               90 NUMDISP = N - INITSS + 1                                          OCRVFIT   140
                  IOCF(41)=N                                                        25MAY76    16
175               RETURN                                                           OCRVFIT   141
C                                                                                   3MARCH     20
C                    ERROR IN AITKEN CALL - RETURN WITH NO FIT                      3MARCH     21
C                                                                                   3MARCH     22
              100 NCR=(ICURVE-1)*2+1                                                3MARCH     23
180               CALL GISSAS (IOCF(NCR),IOCF(NCR+1))                               3MARCH     24
                  IOCF(NCR)=IOCF(NCR+1)=0                                           3MARCH     25
                  IF (ICOATAB(ICURVE,4,2).NE.0) CALL OMRL8D (ICOATAB(ICURVE,4,2))   3MARCH     26
                  ICOATAB(ICURVE,4,2)=0                                             3MARCH     27
                  CALL OMSET (IOFIT,ICOATAB(I,4,1),0)                               3MARCH     28
185               CALL ERROR# (1.,3.5,41HINDEPENDENT VARIABLE ERROR WITH CURVE FIT, 3MARCH     29
                 1     41,7HOCRVFIT,6HINFORM)                                       3MARCH     30
                  RETURN                                                            3MARCH     31
                  END                                                               OCRVFIT   142
```

350

```
      SUBROUTINE DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NPLOT)                ADOPTPRE    1
C                                                                              DISCRVE     3
C         THIS ROUTINE DISPLAYS THE CURVE(SET OF POINTS) INDICATED BY          DISCRVE     4
C         "ICRVPTR". THE PRIMARY INPUTS TO THE ROUTINE ARE THE POINTER         DISCRVE     5
C         TO THE CURVE HEAD AND THE PLOT SCALE DATA. PRIMARY OUTPUTS           DISCRVE     6
C         FROM THE ROUTINE ARE:                                                DISCRVE     7
C              ISYMBOL - THE SYMBOL USED TO DISTINGUISH THIS CURVE FROM        DISCRVE     8
C                        OTHER CURVES ON THE PLOT.                             DISCRVE     9
C               INITSS - THE DISPLAY ADDRESSES FOR THE CURVE ARE STORED       DISCRVE    10
C                        IN ARRAY IDDAD. IDDAD(INITSS) CONTAINS THE            DISCRVE    11
C                        DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED         DISCRVE    12
C                        TO EXHIBIT THE CURVE.(INITSS-INITIAL SUBSCRIPT)       DISCRVE    13
C              NUMDISP - NUMDISP CONTAINS THE NUMBER OF DISPLAYS               DISCRVE    14
C                        REQUIRED TO EXHIBIT THE CURVE.                        DISCRVE    15
C              NPLOT   - NPLOT CONTAINS THE NUMBER OF THE                      ADOPTPRE    2
C                        PLOT IN THE TABLE                                     ADOPTPRE    3
C         INITSS AND NUMDISP ARE NEEDED TO ERASE THE CURVE. THEY WILL BE       DISCRVE    16
C         STORED IN THE CURVE DISPLAY ACCESS TABLE.                            DISCRVE    17
C                                                                              DISCRVE    18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                           /AXSCALE/   2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,         /CRVBEAD/   2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                        /CRVBEAD/   3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),        /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),                 6JULY77     2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPN        1SEPT       1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                         CY6777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                        6JULY77
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                       /PARMSV/    2
      COMMON/POINTS/X(248),Y(248),NPEC(248)                                    /POINTS/    2
      COMMON /SYMTAB/ ISYMTAB(20)                                              /SYMTAB/    2
      DIMENSION MESG(7)                                                        9JULY76     1
C                                                                              DISCRVE    28
C         RECORD THE INITIAL VALUE OF THE IDDAD SUBSCRIPT.                     DISCRVE    29
C                                                                              DISCRVE    30
      INITSS=(NPLOT-1)*4+1                                                     ADOPTPRE    4
      NN=INITSS-1                                                              ADOPTPRE    5
C                                                                              DISCRVE    32
C         GET THE NUMBER OF POINTS ON THE CURVE, THE CURVE SYMBOL              DISCRVE    33
C         NUMBER, THE RECORD BEAD POINTER, THE X-COORDINATE BEAD               DISCRVE    34
C         POINTER, AND THE Y-COORDINATE BEAD POINTER FROM THE                  DISCRVE    35
C         CURVE BEAD.                                                          DISCRVE    36
C                                                                              DISCRVE    37
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                       DISCRVE    38
      CALL DMGET(ICSYM,ICRVPTR,ISYMBOL)                                        DISCRVE    39
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                        DISCRVE    40
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                         DISCRVE    41
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                         DISCRVE    42
C                                                                              DISCRVE    43
C         IF THE NUMBER OF POINTS EXCEEDS THE PROGRAMMED LIMIT(248),           DISCRVE    44
C         DISPLAY THE FIRST 248 POINTS ONLY.                                   DISCRVE    45
C                                                                              DISCRVE    46
      IF(NPOINTS .LE. 248) GO TO 10                                            DISCRVE    47
      NPOINTS=248                                                              DISCRVE    48
      CALL DMSET(ICNPTS,ICRVPTR,NPOINTS)                                       DISCRVE    49
      CALL ERRORM(0.,7.,62HPOINTS LIMIT EXCEEDED. THE FIRST 248 POINTS W       DISCRVE    50
     XILL BE DISPLAYED.,62,7HDISCRVE,6HINFORM)                                 DISCRVE    51
C                                                                              DISCRVE    52
```

```
      C         CONVERT THE CURVE SYMBOL NUMBER INTO A CURVE SYMBOL.
      C
   50    10 ISYMBOL = ISYMTAB(ISYMBOL)
      C
      C         MOVE THE CONTENTS OF THE RECORD NUMBER BEAD, THE X-COORDINATE
      C         BEAD AND THE Y-COORDINATE BEAD INTO CORE. THE RECORD NUMBER
      C         BEAD IS A STATUS BEAD FOR THE POINTS IN THE OTHER TWO BEADS.
   65 C              ACTIVE - THE RECORD NUMBER IS POSITIVE OR SHADDED. THE
      C                       RECORD NUMBER IS SHADDED IF AND ONLY IF THE
      C                       POINT HAS BEEN ADDED DURING AN EDIT.
      C            INACTIVE - THE RECORD NUMBER IS NEGATIVE. A POINT IS
      C                       INACTIVE IF AND ONLY IF IT HAS BEEN DELETED
   70 C                       DURING AN EDIT.
      C
           KOMPCOD = 70C00000001B + NPOINTS * 8**6
           CALL DMGET(KOMPCOD,IRECPTR,NREC)
           CALL DMGET(KOMPCOD,IXPTR,X)
   75      CALL DMGET(KOMPCOD,IYPTR,Y)
      C
      C         LOOP THROUGH THE POINTS. IF A POINT IS INACTIVE OR NOT ON THE
      C         PLOT, SKIP IT. OTHERWISE, CONVERT THE COORDINATES TO DGUS AND
      C         PACK THE DESCRIPTION OF THE POINT INTO THE DESCRIPTION BUFFER.
   80 C
      12 DO 20 I=1,NPOINTS                                              9JULY76    2
           IF(NREC(I) .LT. 0) GO TO 20                                  DISCRVE   77
           IF ((X(I).LT.XMIN).OR.(X(I).GT.XMAX))GO TO 20                30JULY76   1
           IF ((Y(I).LT.YMIN).OR.(Y(I).GT.YMAX))GO TO 20                30JULY76   2
   85      IX=IFIX(((X(I)-XMIN)*DGUPIN)/XINCR+XORDGU-6)                 5OCT       1
           IY=IFIX(((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU - 11.5)      DISCRVE   81
           CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                    DISCRVE   82
           CALL GUAN(ISYMBOL,1,IBUF,NBYTE,MBYTE)                        DISCRVE   83
           IF(NBYTE .LT. 312) GO TO 20                                  DISCRVE   84
   90 C                                                                 DISCRVE   85
      C         THE DESCRIPTION BUFFER IS FULL. FLUSH IT.               DISCRVE   86
      C                                                                 DISCRVE   87
           NN=NN+1                                                      DISCRVE   88
           CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                   DISCRVE   89
   95      20 CONTINUE                                                  DISCRVE   90
      C                                                                 DISCRVE   91
      C         ALL POINTS HAVE BEEN CONSIDERED. IS THE DESCRIPTION     DISCRVE   92
      C         BUFFER EMPTY? IF NOT, FLUSH IT.                         DISCRVE   93
      C                                                                 DISCRVE   94
  100      IF(NBYTE .EQ. 0) GO TO 30                                    DISCRVE   95
           NN=NN+1                                                      DISCRVE   96
           CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                   DISCRVE   97
      30 NUMDISP = NN - INITSS + 1                                      DISCRVE   98
           RETURN                                                       DISCRVE   99
  105      END                                                          DISCRVE  100
```

```
      FUNCTION P(Z,C,N)                                                 P       2
C                                                                       P       3
C         THIS ROUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES P       4
C         POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".          P       5
C                                                                       P       6
C         Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                P       7
C         C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.   P       8
C         N - I - THE NUMBER OF COEFFICIENTS.                           P       9
C                                                                       P      10
      DIMENSION C(N)                                                    P      11
      NDEG = N - 1                                                      P      12
      P=C(1)                                                            18JUNE76 21
      DO 10 I=2,N                                                       18JUNE76 22
      P=P+C(I)*Z**(I-1)                                                 18JUNE76 23
   10 CONTINUE                                                          P      16
      RETURN                                                            P      17
      END                                                               P      18
```

```
      SUBROUTINE REPLOF (NCURVE,ICRVPTR,IFITPTR,IFLAG)                  20SEPT     8
C                                                                       REPLOF     3
C     THIS ROUTINE REPLOTS THE CURVE FITS OF THE                        REPLOF     4
C     CURVES WHICH ARE MARKED "INACTIVE".                               REPLOF     5
C                                                                       REPLOF     6
C                                                                       REPLOF     7
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(40),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIFM 1SEPT      1
      COMMON /FITBEAD/ IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA             REPLOF    10
      DIMENSION X(249),Y(248),FP(248)                                   REPLOF    11
C                                                                       REPLOF    12
C     QUERY THE CURVE AND CURVE FIT HEADS FOR THE CALL TO DCRVFIT       REPLOF    13
C                                                                       REPLOF    14
      IF (IFLAG.LE.0) GO TO 20                                          20SEPT     9
      IFITPTR=ICDATAB(NCURVE,4,2)                                       REPLOF    15
      ICRVPTR=ICDATAB(NCURVE,4,1)                                       REPLOF    16
C                                                                       REPLOF    17
   20 CALL DMGET (IFTYPE,IFITPTR,ITYPE)                                 13SEPT     7
      CALL DMGET (ICINDEP,ICRVPTR,IVAR)                                 REPLOF    19
      CALL DMGET (IFIVVFP,IFITPTR,XINDFP)                               REPLOF    20
      CALL DMGET (IFIVVLP,IFITPTR,XINDLP)                               REPLOF    21
      CALL DMGET (IFIVI,IFITPTR,XINC)                                   REPLOF    22
C                                                                       REPLOF    23
      IF (ITYPE.EQ.6HAITKEN) GO TO 60                                   20SEPT
      CALL DMGET (IFNPARA,IFITPTR,NFTPA)                                REPLOF
      KOMPCOD=7C0000000063 + SHIFT(NFTPA,18)                            REPLOF    25
      CALL DMGET (KOMPCOD,IFITPTR,FP)                                   REPLOF    26
C                                                                       REPLOF    27
C     IF FIT IS LEAST SQUARES CALL DCRVFIT                              REPLOF    28
C     SPLINE NEEDS MORE INFORMATION                                     REPLOF    29
C                                                                       REPLOF    30
      IF (ITYPE.EQ.6HSPLINE) GO TO 50                                   REPLOF    31
      CALL DCRVFIT (ITYPE,IVAR,XDUM,YDUM,NDUM,FP,NFTPA,XINDFP,XINDLP,   REPLOF    32
     1    XINC,INITSS,NUMDISP,NCURVE)                                   REPLOF    33
      GO TO 100                                                         REPLOF    34
C                                                                       REPLOF    35
C     SPLINE FIT -- RETRIEVE EXTRA INFORMATION                          REPLOF    36
C                                                                       REPLOF    37
   50 KOMPCOD=KOMPCOD+NFTPA                                             REPLOF    38
      CALL DMGET (KOMPCOD,IFITPTR,X)                                    REPLOF    39
      KOMPCOD=KOMPCOD+NFTPA                                             REPLOF    40
      CALL DMGET (KOMPCOD,IFITPTR,Y)                                    REPLOF    41
C                                                                       REPLOF    42
      CALL DCRVFIT (ITYPE,IVAR,X,Y,NFTPA,FP,NFTPA,XINDFP,XINDLP,        REPLOF    43
     1    XINC,INITSS,NUMDISP,NCURVE)                                   REPLOF    44
      GO TO 100                                                         20SEPT    11
C                                                                       20SEPT    12
C         AITKEN FIT -- RETRIEVE X AND Y ARRAYS                         20SEPT    13
C                                                                       20SEPT    14
   60 CALL DMGET (IFNPARA,IFITPTR,NPOINTS)                              20SEPT    15
      CALL DMGET (IFNPARA+1,IFITPTR,NFTPA)                              5OCT       5
      KOMPCOD=7C0000000073+SHIFT(NPOINTS,18)                            20SEPT    17
      CALL DMGET (KOMPCOD,IFITPTR,X)                                    20SEPT    18
      CALL DMGET (KOMPCOD+NPOINTS,IFITPTR,Y)                            20SEPT    19
```

```
      C                                                                  20SEPT     20
           CALL DCRVFIT (ITYPE,IVAR,X,Y,NPOINTS,FP,NFTPA,XINOFP,XINOLP,  20SEPT     21
     6C   1     XINC,INITSS,NUMDISP,NCURVE)                              20SEPT     22
      C                                                                  REPLCF     45
      C     UPDATE CURVE DISPLAY TABLE                                   REPLCF     46
      C                                                                  REPLCF     47
        10C CALL UCDATAB (NCURVE,2,6HACTIVE,INITSS,NUMDISP,IFITPTR)      REPLCF     48
     65     RETURN                                                       REPLCF     49
            END                                                          REPLCF     50
```

```
      SUBROUTINE REPLOT                                                 REPLOT     2
C                                                                       REPLOT     3
C         THIS ROUTINE SEARCHES THROUGH THE CURVE DISPLAY ACCESS TABLE  REPLOT     4
C         AND REDISPLAYS ALL CURVES LISTED "INACTIVE" IN THE TABLE.     REPLOT     5
C                                                                       REPLOT     6
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/  2
      COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               ICAXIS(4),IDTITLE(4),IOCO(23),IOMENU(20),         6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
C                                                                       REPLOT     9
C         LOOP THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE. IF A REPLOT     10
C         CURVE IS ACTIVE, INCREASE THE IODAD INDEX(NN) BY THE NUMBER OF REPLOT     11
C         ELEMENTS OF IODAD REQUIRED TO DISPLAY IT AND THEN GO TO THE   REPLOT     12
C         NEXT ONE. IF A CURVE IS INACTIVE, THEN DISPLAY IT AND ITS     REPLOT     13
C         DESCRIPTION AND UPDATE THE CURVE DISPLAY ACCESS TABLE.        REPLOT     14
C                                                                       REPLOT     15
      CALL ORIGIN(-5.,-3.)                                              5MAY76     7
      NN = 0                                                            REPLOT     16
      DO 100 I=1,NCIP                                                   3JUNE76    1
      IF(ICDATAB(I,1,1) .EQ. 8HINACTIVE) GO TO 10                       14APR76    5
      NN = NN + ICDATAB(I,3)                                            REPLOT     19
      GO TO 30                                                          REPLOT     20
   10 CALL DISCRVE (ICDATAB(I,4,1),ISYMBOL,INITSS,NUMDISP,I)            ADFTPRE    7
      IF(NUMDISP .GT. 0) GO TO 20                                       REPLOT     22
C                                                                       REPLOT     23
C         THIS CODE IS HERE TO COVER THE POSSIBILITY THAT EVERY POINT ON REPLOT     24
C         A CURVE HAS BEEN DELETED. THE CURVE IS ACTIVE BUT EMPTY.      REPLOT     25
C                                                                       REPLOT     26
      NN = NN + 1                                                       REPLOT     27
      IODAD(NN) = 0                                                     REPLOT     28
      CALL DCRVDES (ICDATAB(I,4,1),ISYMBOL,I)                           25MAY76    4
      CALL UCDATAB(I,1,6HACTIVE,INITSS,1,ICDATAB(I,4,1))                14APR76    6
      GO TO 30                                                          REPLOT     31
   20 CALL DCRVDES (ICDATAB(I,4,1),ISYMBOL,I)                           25MAY76    5
      CALL UCDATAB(I,1,6HACTIVE,INITSS,NUMDISP,ICDATAB(I,4,1))          14APR76    7
   30 IF (ICDATAB(I,4,2).LE.0) GO TO 100                                3JUNE76    2
      IF (ICDATAB(I,1,2).EQ.6HACTIVE) GO TO 100                         3JUNE76    3
      CALL REPLOF (I,I1,I2,1)                                           20SEPT     23
  100 CONTINUE                                                          3JUNE76    5
      RETURN                                                            REPLOT     35
      END                                                               REPLOT     36
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C  -  -  A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE    SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 8 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 8                                          SPLIN2    27
       KK = K                                                            SPLIN2
       GO TO 120                                                         SPLIN2
    8 CONTINUE                                                           SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN                                                             SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1       1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3    SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
      RETURN                                                             SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
      OVERLAY(E,0)                                                      CRVEFIT     2
      PROGRAM CRVEFIT                                                   CRVEFIT     3
C                                                                       CRVEFIT     4
C         "CRVEFIT" IDENTIFIES THE CURVES WHICH ARE TO BE CURVE FIT,    CRVEFIT     5
C         ERASES THE CURVES WHICH ARE NOT TO BE FIT, IDENTIFIES THE     CRVEFIT     6
C         TYPE OF CURVE FIT, AND CALLS THE ROUTINE WHICH WILL EXECUTE   CRVEFIT     7
C         THE SELECTED CURVE FIT.                                       CRVEFIT     8
C                                                                       CRVEFIT     9
      COMMON/BLOCK/IBLOCK(50)                                           /BLOCK/     2
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),         6JULY77     2
     X               IDINSTR(19),IDDATA(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT       1
      COMMON /GP/ IBUF(128),NCON,N8YTE,M8YTE,IGN,ISIN,                  CYBER77     1
     X     ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77     3
      DIMENSION IODTSTR(16), IODCSTR(16)                                CRVEFIT    13
      EQUIVALENCE (IODTSTR,IBLOCK), (IODCSTR,IBLOCK(17))                CRVEFIT    14
C                                                                       CRVEFIT    15
C         CLEAR THE QUEUES AND CHANGE THE PICK TYPE OF EACH OF          CRVEFIT    16
C         THE CURVE DESCRIPTIONS FROM "IGNORE" TO "STRING".             CRVEFIT    17
C                                                                       CRVEFIT    18
   10 CALL GICLR(NCON)                                                  CRVEFIT    19
      CALL CPTYPE(ICCD,20,1,NCIP,ISTR)                                  CRVEFIT    20
C                                                                       CRVEFIT    21
C         DISPLAY THE CURVE FIT MENU AND INSTRUCTIONS.                  CRVEFIT    22
C         WAIT FOR A BUTTON PICK FROM THE FOLLOWING MENU.               CRVEFIT    23
C              71 - LEAST SQUARES                                       CRVEFIT    24
C              72 - SPLINE                                              CRVEFIT    25
C              73 - STRAIGHT LINE                                       CRVEFIT    26
C              76 - RETURN TO MASTER CONTROL MENU                       CRVEFIT    27
C             254 - STOP                                                14APR76   59
C         "ALL" IS DISPLAYED AS A SINGLE PICK.                          CRVEFIT    29
C         THE CURVE DESCRIPTIONS ARE STRING PICKS.                      CRVEFIT    30
C                                                                       CRVEFIT    31
   20 CALL CFMENU                                                       CRVEFIT    32
      CALL CFINST                                                       CRVEFIT    33
      CALL ORIGIN(-5.0,-3.0)                                            CRVEFIT    34
      CALL ANNOTA(11.0,-3.0,3HALL,3,IOBLOCK(1),ISIN,-0,-0,-0)           CRVEFIT    35
   30 CALL GIBUT(0,NCON,IODTBUT,IODCBUT)                                CRVEFIT    36
      IF (IODCBUT.EQ.76) GO TO 80                                       18JUNE76   10
      IF(IODCBUT .EQ. 254) CALL STOP                                    14APR76    60
C                                                                       CRVEFIT    39
C         CHECK TO SEE IF "ALL" WAS SINGLE PICKED.                      CRVEFIT    40
C         IF SO, SKIP THE TEST FOR STRING PICKS.                        CRVEFIT    41
C                                                                       CRVEFIT    42
      CALL GIFID(NCON,IODTSIN)                                          CRVEFIT    43
      IF(IODTSIN .EQ. 0) GO TO 40                                       CRVEFIT    44
      CALL GICLR(NCON)                                                  CRVEFIT    45
      GO TO 80                                                          CRVEFIT    46
C                                                                       CRVEFIT    47
C         "ALL" WAS NOT SINGLE PICKED. RETRIEVE THE IDENTITY            CRVEFIT    48
C         OF THE CURVE DESCRIPTIONS WHICH WERE STRING PICKED.           ORVEFIT    49
C                                                                       CRVEFIT    50
   40 NSTRP=10                                                          25MAY76    12
      CALL GIFSID(NCON,NSTRP,IODTSTR,IODCSTR)                           25MAY76    13
      IF(NSTRP .GT. 0) GO TO 50                                         CRVEFIT    52
      CALL ERROFM(-5.0,+4.5,47HYOU DID NOT INDICATE WHICH CURVES TO CURV CRVEFIT    53
     XE FIT.,47,7HCRVEFIT,6HINFORM)                                     CRVEFIT    54
```

```
                GO TO 30                                                        CRVEFIT   55
      C                                                                         CRVEFIT   56
   60 C         LOOP THROUGH THE CURVES IN THE CURVE DISPLAY ACCESS TABLE. IF   CRVEFIT   57
      C         THE "I-TH" CURVE IN THE TABLE MATCHES ONE OF THE STRING PICKS,  CRVEFIT   58
      C         THEN GO AHEAD TO THE "(I+1)-ST" CURVE IN THE TABLE. IF THE      CRVEFIT   59
      C         "I-TH" CURVE IN THE TABLE DOES NOT MATCH ANY OF THE STRING      CRVEFIT   60
      C         PICKS, THEN ERASE IT AND CHANGE ITS STATUS IN THE CURVE         CRVEFIT   61
   65 C         DISPLAY ACCESS TABLE TO INACTIVE.                               CRVEFIT   62
      C                                                                         CRVEFIT   63
        50 DO 70 I=1,NCIF                                                       CRVEFIT   64
           DO 60 J=1,NSTRP                                                      CRVEFIT   65
           IF(ICDCSTR(J) .EQ. I) GO TO 70                                       CRVEFIT   66
   70      60 CONTINUE                                                          CRVEFIT   67
           CALL TERASE(ICDAC(ICDATAB(I,2,1)),ICDATAB(I,3,1))                    25MAY76   14
           ICDATAB(I,1,1) = 8HINACTIVE                                          14APR76   61
           CALL GIERAS (ICDCD(I))                                               3JUNE76    6
           IF (ICDATAB(I,4,2).LE.0) GO TO 70                                    3JUNE76    7
   75      CALL TERASE (ICDCF(ICDATAB(I,2,2)),ICDATAB(I,3,2))                   3JUNE76    8
           ICDATAB(I,1,2)=8HINACTIVE                                            3JUNE76    9
        70 CONTINUE                                                             CRVEFIT   70
      C                                                                         CRVEFIT   71
      C         ERASE THE CURVE FIT MENU AND INSTRUCTIONS. DO NOT               CRVEFIT   72
   80 C         CHANGE THE STATUS OF "ALL" OR THE CURVE DESCRIPTIONS.           CRVEFIT   73
      C                                                                         CRVEFIT   74
359     80 CALL TERASE(ICMENU,ICMENU(13))                                       CRVEFIT   75
           CALL TERASE(ICINSTR,ICINSTR(19))                                     CRVEFIT   76
      C                                                                         CRVEFIT   77
   85 C         IDENTIFY AND EXECUTE THE BUTTON PICK.                           CRVEFIT   78
      C                                                                         CRVEFIT   79
           IF(ICDCBUT .EQ. 71) GO TO 90                                         CRVEFIT   80
           IF(ICDCBUT .EQ. 72) GO TO 100                                        CRVEFIT   81
           IF(ICDCBUT .EQ. 73) GO TO 110                                        CRVEFIT   82
   90      IF (ICDCBUT.NE.76) GO TO 20                                          9JULY76   19
           CALL GIERAS (ICBLOCK(1))                                             9JULY76   20
           CALL STOFIT                                                          DELRAD     8
           CALL AETSKC (7HQTRBACK)                                              9JULY76   21
      C                                                                         CRVEFIT   84
   95 C         LEAST SQUARES WAS SELECTED.                                     CRVEFIT   85
      C                                                                         CRVEFIT   86
        90 CALL AETSKC (7HLEASTSQ)                                              INHALE    63
           CALL AETSKC (5HCONCF)                                                INHALE    64
           GO TO 10                                                             CRVEFIT   89
  100      •                                                                    CRVEFIT   90
      C         SPLINE WAS SELECTED.                                            CRVEFIT   91
      C                                                                         CRVEFIT   92
       100 CALL AETSKC (6HSPLINE)                                               INHALE    65
           CALL AETSKC (5HCONCF)                                                INHALE    66
  105      GO TO 10                                                             CRVEFIT   95
      C                                                                         CRVEFIT   96
      C         STRAIGHT LINE WAS SELECTED.                                     CRVEFIT   97
      C                                                                         CRVEFIT   98
       110 CALL AETSKC (6HSTRAIT)                                               INHALE    67
  110      CALL AETSKC (5HCONCF)                                                INHALE    68
           GO TO 10                                                             CRVEFIT  101
           END                                                                  CRVEFIT  102
```

```
      SUBROUTINE CFINST                                                 CFINST     2
C                                                                       CFINST     3
C         THIS ROUTINE DISPLAYS THE INSTRUCTIONS                        CFINST     4
C         WHICH ACCOMPANY THE CURVE FIT MENU.                           CFINST     5
C                                                                       CFINST     6
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/ 2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),         6JULY77    2
     Y               IDINSTR(19),ICDATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT    1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      CALL ORIGIN(-4.38,-5.0)                                           CFINST     9
      CALL SQUARE(0.,-3.,3.38,3.,IDINSTR(1),IGN,-0)                     CFINST    10
      CALL ANNOTA(.97,.1,12HINSTRUCTIONS,12,IDINSTR(2),IGN,-0,-0,-0)    CFINST    11
      CALL ANNOTA(.25,-.5,21H1.TO CURVE FIT ALL OF,21,IDINSTR(3),IGN,   CFINST    12
     X            -0,-0,-0)                                            CFINST    13
      CALL ANNOTA(.25,-.75,13H  THE CURVES.,13,IDINSTR(4),IGN,-0,-0,-0) CFINST    14
      CALL ANNOTA(.25,-1.0,16H   A.PICK "ALL".,16,IDINSTR(5),IGN,-0,-0, CFINST    15
     X            -0)                                                  CFINST    16
      CALL ANNOTA(.25,-1.25,23H    B.PICK THE FIT TYPE.,23,IDINSTR(6),  CFINST    17
     X            IGN,-0,-0,-0)                                        CFINST    18
      CALL ANNOTA(.25,-1.75,22H2.TO CURVE FIT SOME OF,22,IDINSTR(7),    CFINST    19
     X            IGN,-0,-0,-0)                                        CFINST    20
      CALL ANNOTA(.25,-2.0,13H  THE CURVES.,13,IDINSTR(8),IGN,-0,-0,-0) CFINST    21
      CALL ANNOTA(.25,-2.25,20H   A.STRING PICK THE,20,IDINSTR(9),IGN,  CFINST    22
     X            -0,-0,-0)                                            CFINST    23
      CALL ANNOTA(.25,-2.50,24H     CURVE DESCRIPTIONS.,24,IDINSTR(10), CFINST    24
     X            IGN,-0,-0,-0)                                        CFINST    25
      CALL ANNOTA(.25,-2.75,23H    B.PICK THE FIT TYPE.,23,IDINSTR(11), CFINST    26
     X            IGN,-0,-0,-0)                                        CFINST    27
      IDINSTR(19) = 11                                                  CFINST    28
      RETURN                                                            CFINST    29
      END                                                               CFINST    30
```

```
      SUBROUTINE CFMENU                                                  CFMENU      2
C                                                                        CFMENU      3
C         THIS ROUTINE DISPLAYS THE CURVE FIT MENU.                      CFMENU      4
C                                                                        CFMENU      5
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IDSPEC(19),IOCF(41),  /DISPLAY/  2
     X               IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),          6JULY77     2
     X               IOINSTR(19),ICOATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT       1
      COMMON /GP/ IBUF(126),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYBTTT      1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77     3
      CALL ORIGIN(0.,-5.)                                                CFMENU      8
      CALL SQUARE(0.,-2.25,3.98,2.25,IOMENU(1),IGN,-0)                   CFMENU      9
      CALL ANNOTA(1.15,.1,14HCURVE FIT MENU,14,IOMENU(2),IGN,-0,-0,-0)   CFMENU     10
      CALL ANNOTA(.25,-.5,13HLEAST SQUARES,13,IOMENU(3),IBUT,71,-0,-0)   CFMENU     11
      CALL ANNOTA(.25,-1.0,6HSPLINE,6,IOMENU(4),IBUT,72,-0,-0)           CFMENU     12
      CALL ANNOTA(.25,-1.5,13HSTRAIGHT LINE,13,IOMENU(5),IBUT,73,-0,-0)  CFMENU     13
      CALL ANNOTA(.25,-2.0,29HRETURN TO MASTER CONTROL MENU,29,          CFMENU     14
     X            IOMENU(6),IBUT,76,-0,-0)                               CFMENU     15
      IOMENU(13) = 6                                                     CFMENU     16
      RETURN                                                             CFMENU     17
      END                                                                CFMENU     18
```

```
      SUBROUTINE STOFIT                                                 STOFIT     2
C                                                                       STOFIT     3
C     STOFIT STORES THE FIT HEAD POINTER IN THE CURVE HEAD              STOFIT     4
C                                                                       STOFIT     5
      COMMON/CRVHEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVHEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVHEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(91),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
C                                                                       STOFIT     8
      DO 50 J=1,NCIP                                                    STOFIT     9
      IF (ICDATAB(J,4,2).EQ.0) GO TO 50                                 STOFIT    10
      IF (ICDATAB(J,4,1).EQ.0) GO TO 50                                 STOFIT    11
      CALL DMSET (ICFIT,ICDATAB(J,4,1),ICDATAB(J,4,2))                  STOFIT    12
   50 CONTINUE                                                          STOFIT    13
      RETURN                                                            STOFIT    14
      END                                                               STOFIT    15
```

```
      SUBROUTINE CPTYPE(IDDAD,L,M,N,NEWTYPE)                                CPTYPE     2
C                                                                           CPTYPE     3
C         THIS SOUTINE CHANGES THE PICK TYPE OF SOME OF THE DISPLAYS        CPTYPE     4
C         WHOSE ADDRESSES ARE STORED IN ARRAY IDDAD. THE PICK TYPE OF       CPTYPE     5
C         DISPLAYS IDDAD(M) THROUGH IDDAD(N) IS CHANGED FROM ITS CURRENT    CPTYPE     6
C         TYPE TO NEWTYPE. "L" IS THE LENGTH OF THE ARRAY IDDAD.            CPTYPE     7
C                                                                           CPTYPE     8
      DIMENSION IDDAD(L)                                                    CPTYPE     9
      DO 10 I=M,N                                                           CPTYPE    10
      CALL GIMOVE(-0,-0,-0,IDDAD(I),NEWTYPE)                                CPTYPE    11
   10 CONTINUE                                                              CPTYPE    12
      RETURN                                                                CPTYPE    13
      END                                                                   CPTYPE    14
```

```
      SUBROUTINE DCRVFIT(ITYPE,IV,X,Y,NPOINTS,C,NFP,TMIN,TMAX,TINCR,     DCRVFIT     2
     X          INITSS,NUMDISP,ICURVE)                                   28MAY76     1
C                                                                       DCRVFIT     4
C     THIS SOUTINE DISPLAYS THE CURVE FIT DEFINED BY PARAMETERS ONE     DCRVFIT     5
C     THROUGH SEVEN OVER THE INTERVAL DEFINED BY PARAMETERS EIGHT       DCRVFIT     6
C     AND NINE.                                                         DCRVFIT     7
C               SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND             13SEPT     18
C               6HAITKEN FOR A STRAIGHT FIT.                            13SEPT     19
C               SQUARES FIT AND 6HSPLINE FOR A SPLINE FIT.              DCRVFIT     9
C          IV - I - THE INDEPENDENT VARIABLE; EITHER 1HX OR 1HY.        DCRVFIT    10
C           X - I - THE ARRAY OF X-COORDINATES OF THE POINTS FOR        DCRVFIT    11
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      DCRVFIT    12
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     DCRVFIT    13
C           Y - I - THE ARRAY OF Y-COORDINATES OF THE POINTS FOR        DCRVFIT    14
C               WHICH THE CURVE FIT WAS CONSTRUCTED. THIS ARRAY IS      DCRVFIT    15
C               SUPERFLOUS FOR A LEAST SQUARES FIT.                     DCRVFIT    16
C     NPOINTS - I - THE NUMBER OF POINTS WHOSE COORDINATES              DCRVFIT    17
C               ARE IN ARRAYS "X" AND "Y".                              DCRVFIT    18
C           C - I - THE ARRAY OF CURVE FIT PARAMETERS; POLYNOMIAL       DCRVFIT    19
C               COEFFICIENTS FOR A LEAST SQUARES FIT, DERIVATIVES       DCRVFIT    20
C               AT EACH POINT FOR A SPLINE FIT.                         DCRVFIT    21
C         NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF         DCRVFIT    22
C               POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT,        DCRVFIT    23
C               THE NUMBER OF POINTS FOR A SPLINE FIT.                  DCRVFIT    24
C        TMIN - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      DCRVFIT    25
C               THE DISPLAY OF THE CURVE FIT IS TO START.               DCRVFIT    26
C        TMAX - I - THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH      DCRVFIT    27
C               THE DISPLAY OF THE CURVE FIT IS TO STOP. (TMIN,TMAX)    DCRVFIT    28
C               IS THE INTERVAL OVER WHICH THE CURVE FIT IS TO BE       DCRVFIT    29
C               DISPLAYED.                                              DCRVFIT    30
C       TINCR - I - THE INCREMENT OF THE INDEPENDENT VARIABLE PER       DCRVFIT    31
C               TWENTIETH OF AN INCH. THE CURVE FIT WILL BE             DCRVFIT    32
C               APPROXIMATED BY TWENTY STRAIGHT LINES PER INCH.         DCRVFIT    33
C      INITSS - O - THE DISPLAY ADDRESSES OF THE CURVE FIT ARE          DCRVFIT    34
C               STORED IN ARRAY "IDCF". IDCF(INITSS) CONTAINS THE       DCRVFIT    35
C               DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED TO        DCRVFIT    36
C               EXHIBIT THE CURVE FIT.                                  DCRVFIT    37
C     NUMDISP - O - NUMDISP CONTAINS THE NUMBER OF DISPLAYS             DCRVFIT    38
C               REQUIRED TO EXHIBIT THE CURVE FIT.                      DCRVFIT    39
C                                                                       DCRVFIT    40
C     A 64 WORD DESCRIPTION BUFFER WILL HOLD THE DESCRIPTION BYTES      DCRVFIT    41
C     FOR 101 LINE SEGMENTS.                                            DCRVFIT    42
C        NUMBER      TYPE      BYTES                                    DCRVFIT    43
C          OF         OF       PER       TOTAL                         DCRVFIT    44
C        CALLS      CALL      CALL      BYTES                          DCRVFIT    45
C        *******    ******    *******   *******                        DCRVFIT    46
C           1       GURSET       3          3                         DCRVFIT    47
C           1       GUSEGI       4          4                         DCRVFIT    48
C         101       GUSEG        3        303                         DCRVFIT    49
C           1       GIDISP      10         10                         DCRVFIT    50
C                                          320---TOTAL                DCRVFIT    51
C     SINCE ANY CURVE FIT CAN BE DISPLAYED WITH AT MOST 100 LINE        DCRVFIT    52
C     SEGMENTS(NINE INCHES), EACH CURVE FIT CAN BE DISPLAYED WITH       DCRVFIT    53
C     AT MOST TWO BUFFERS.                                              DCRVFIT    54
C                                                                       DCRVFIT    55
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/   2
      COMMON/DISPLAY/IDSTOP,IDDAD(30),NU,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/   2
```

```
                     X                IOAXIS(4),IOTITLE(4),IOCO(23),IOMENU(20),                6JULY77    2
                     Y                IOINSTR(19),ICDATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM       1SEPT      1
   60                 COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                         OY0777     1
                     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                       6JULY77    3
                      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                      /PARMSV/   2
                      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,        /CRVBEAD/   2
                     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                       /CRVBEAD/   3
   65                 DIMENSION TEMP(14)                                                      13SEPT     20
                      DIMENSION X(NPOINTS),Y(NPOINTS),C(NFP)                                  OCRVFIT    60
                C                                                                             OCRVFIT    61
                C         RECORD THE INITIAL VALUE OF THE "IOCF" SUBSCRIPT.                   OCRVFIT    62
                C                                                                             OCRVFIT    63
   70                 N=(ICURVE-1)*2                                                          28MAY76    2
                      INITSS = N + 1                                                          OCRVFIT    65
                C                                                                             OCRVFIT    66
                      IF(IV .EQ. 1HY) GO TO 50                                                OCRVFIT    67
                      IF(IV .EQ. 1HX) CONTINUE                                                OCRVFIT    68
   75           C                                                                             OCRVFIT    69
                C         INITIALIZE THE DESCRIPTION BUFFER TO THE FIRST POINT ON THE         OCRVFIT    70
                C         CURVE FIT. THE Y-COORDINATE OF THE POINT DEPENDS UPON THE           OCRVFIT    71
                C         TYPE OF CURVE FIT.                                                  OCRVFIT    72
                C                                                                             OCRVFIT    73
   80                 XX = XMIN                                                              OCRVFIT    74
                      XXINCR = XINCR                                                          OCRVFIT    75
                      XXMAX = XMAX                                                            OCRVFIT    76
                    2 IF (ITYPE.EQ.10HLEASTSQUAR) YY=P(XX,C,NFP)                              29APR      2
                      IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)                     OCRVFIT    78
   85                 IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,                20SEPT     76
                     1     YY,TEMP,IER)                                                       13SEPT     22
                      IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX)            29APR      3
                     2     GO TO 5                                                            27MAY      3
                      XX=XX+XXINCR                                                            29APR      5
   90                 IF (XX.GT.XXMAX) GO TO 32                                               29APR      6
                      GO TO 2                                                                 29APR      7
                    5 IF (IER.EQ.2) GO TO 100                                                 29APR      8
                      IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)                   OCRVFIT    79
                      IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)                   OCRVFIT    80
   95                 NBYTE = 0                                                               OCRVFIT    81
                   10 CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                               OCRVFIT    82
                      CALL GUSEGI(IX,IY,ISTYLE,IBUF,NBYTE,MBYTE)                              OCRVFIT    83
                C                                                                             OCRVFIT    84
                C         INCREMENT THROUGH THE INTERVAL AND PACK THE DESCRIPTION BYTES       OCRVFIT    85
  100           C         OF APPROXIMATING LINE SEGMENTS INTO THE DESCRIPTION BUFFER.         OCRVFIT    86
                C                                                                             OCRVFIT    87
                   20 XX = XX + XXINCR                                                        OCRVFIT    88
                      IF(ITYPE .EQ. 10HLEASTSQUAR) YY = P(XX,C,NFP)                           OCRVFIT    89
                      IF(ITYPE .EQ. 6HSPLINE) YY = SPLIN2(XX,NFP,X,Y,C,1)                     OCRVFIT    90
  105                 IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (X,Y,NPOINTS,NFP,XX,                20SEPT     77
                     1     YY,TEMP,IER)                                                       13SEPT     24
                      IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 30                                  14MARCH    17
                      IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 30                                  14MARCH    18
                      IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)                   OCRVFIT    91
  110                 IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)                   OCRVFIT    92
                      CALL GUSEG(IX,IY,IBEAM)                                                 OCRVFIT    93
                C                                                                             OCRVFIT    94
                C         CHECK TO SEE IF THE DESCRIPTION BUFFER IS FULL.                     OCRVFIT    95
                C         IF SO, FLUSH IT. IF NOT, CONTINUE TO FILL IT.                       OCRVFIT    96
```

```
      C                                                                  14APR76   71
            IF(NBYTE .LT. 312) GO TO 30                                  DCRVFIT   97
            N = N + 1                                                    DCRVFIT   98
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                     DCRVFIT   99
            IF(XX .LT. XXMAX) GO TO 10                                   DCRVFIT  100
      C                                                                  DCRVFIT  101
      C        THE DESCRIPTION BUFFER IS NOT FULL. CHECK TO SEE IF THE CURVE  DCRVFIT  102
      C         FIT IS COMPLETE. IF NOT, CONTINUE TO GENERATE THE DESCRIPTION DCRVFIT  103
      C         BYTES FOR THE CURVE FIT. IF SO, DISPLAY THE CURVE FIT.   DCRVFIT  104
      C                                                                  DCRVFIT  105
   30 IF(XX .LT. XXMAX) GO TO 20                                         DCRVFIT  106
   32 IF (NBYTE.EQ.0) GO TO 40                                           29APR     9
            N = N + 1                                                    DCRVFIT  108
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                     DCRVFIT  109
   40 NUMDISP = N - INITSS + 1                                           DCRVFIT  110
            IDCF(41)=N                                                   25MAY76   15
            RETURN                                                       DCRVFIT  111
      C                                                                  DCRVFIT  112
      C        "Y" IS THE INDEPENDENT VARIABLE. COMMENTS                 DCRVFIT  113
      C        ANALOGOUS TO THOSE ABOVE APPLY.                           DCRVFIT  114
      C                                                                  DCRVFIT  115
   50 YY = YMIN                                                          DCRVFIT  116
            YYINCR = YINCR                                               DCRVFIT  117
            YYMAX = YMAX                                                 DCRVFIT  118
   52 IF (ITYPE.EQ.10HLEASTSQUAR) XX=P(YY,C,NFP)                         29APR     10
            IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)          DCRVFIT  120
            IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,     20SEPT    7
           1    XX,TEMP,IER)                                             13SEPT
            IF (YY.GE.YMIN.AND.YY.LE.YMAX.AND.XX.GE.XMIN.AND.XX.LE.XMAX) 29APR     11
           1       GO TO 55                                              29APR     12
            YY=YY+YYINCR                                                 29APR     13
            IF (YY.GT.YYMAX) GO TO 82                                    29APR     14
            GO TO 52                                                     29APR     15
   55 IF (IER.EQ.2) GO TO 100                                            29APR     16
            IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)        DCRVFIT  121
            IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)        DCRVFIT  122
            NBYTE = 0                                                    DCRVFIT  123
   60 CALL GURSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                          DCRVFIT  124
            CALL GUSEGI(IX,IY,ISTYLE,IBUF,NBYTE,MBYTE)                   DCRVFIT  125
   70 YY = YY + YYINCR                                                   DCRVFIT  126
            IF(ITYPE .EQ. 10HLEASTSQUAR) XX = P(YY,C,NFP)                DCRVFIT  127
            IF(ITYPE .EQ. 6HSPLINE) XX = SPLIN2(YY,NFP,Y,X,C,1)          DCRVFIT  128
            IF(ITYPE .EQ. 6HAITKEN) CALL INTERP (Y,X,NPOINTS,NFP,YY,     20SEPT   79
           1    XX,TEMP,IER)                                             13SEPT   28
            IF (YY.LT.YMIN.OR.YY.GT.YMAX) GO TO 80                       14MARCH  21
            IF (XX.LT.XMIN.OR.XX.GT.XMAX) GO TO 80                       14MARCH  22
            IF (IER.EQ.2) GO TO 100                                      3MARCH   19
            IX = IFIX(((XX - XMIN) * DGUPIN)/XINCR + XORDGU + .5)        DCRVFIT  129
            IY = IFIX(((YY - YMIN) * DGUPIN)/YINCR + YORDGU + .5)        DCRVFIT  130
            CALL GUSEG(IX,IY,IBEAM)                                      DCRVFIT  131
            IF(NBYTE .LT. 312) GO TO 90                                  DCRVFIT  132
            N = N + 1                                                    DCRVFIT  133
            CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                     DCRVFIT  134
            IF(YY .LT. YYMAX) GO TO 60                                   DCRVFIT  135
   80 IF(YY .LT. YYMAX) GO TO 70                                         DCRVFIT  136
   82 IF (NBYTE.EQ.0) GO TO 90                                           29APR     17
            N = N + 1                                                    DCRVFIT  138
```

```
                  CALL GIDISP(NCON,IBUF,NBYTE,IDCF(N),IGN)                          DCRVFIT   139
                  NUMDISP = N - INITSS + 1                                          DCRVFIT   140
                  IDCF(41)=N                                                        25MAR76    16
175               RETURN                                                            DCRVFIT   141
            C                                                                       3MARCH     20
            C        ERROR IN AITKEN CALL - RETURN WITH NO FIT                      3MARCH     21
            C                                                                       3MARCH     22
              100 NCR=(ICURVE-1)*2+1                                                3MARCH     23
180               CALL GTESAS (IDCF(NCR),IDCF(NCR+1))                               3MARCH     24
                  IDCF(NCR)=IDCF(NCR+1)=0                                           3MARCH     25
                  IF (ICDATAB(ICURVE,4,2).NE.0) CALL DMRLBD (ICDATAB(ICURVE,4,2))   3MARCH     26
                  ICDATAB(ICURVE,4,2)=0                                             3MARCH     27
                  CALL DMSET (ICFIT,ICDATAB(I,4,1),0)                               3MARCH     28
185               CALL ERROR4 (1.,3.5,41HINDEPENDENT VARIABLE ERROR WITH CURVE FIT, 3MARCH     29
                 1    41,7HDCRVFIT,6HINFORM)                                       3MARCH     30
                  RETURN                                                            3MARCH     31
                  END                                                               DCRVFIT   142
```

```
      FUNCTION P(Z,C,N)                                                 P          2
C                                                                       P          3
C         THIS ROUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES P          4
C         POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".          P          5
C                                                                       P          6
C         Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                P          7
C         C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.   P          8
C         N - I - THE NUMBER OF COEFFICIENTS.                           P          9
C                                                                       P         10
      DIMENSION C(N)                                                    P         11
      NDEG = N - 1                                                      P         12
      P=C(1)                                                            18JUNE76  21
      DO 10 I=2,N                                                       18JUNE76  22
      P=P+C(I)*Z**(I-1)                                                 18JUNE76  23
   10 CONTINUE                                                          P         16
      RETURN                                                            P         17
      END                                                               P         18
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C - - A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE       SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 9 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 9                                          SPLIN2    27
       KK = K                                                            SPLIN2    28
       GO TO 120                                                         SPLIN2    29
    9 CONTINUE                                                           SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN                                                             SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1        1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3   SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
       RETURN                                                            SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
      SUBROUTINE UCDATA3(NROW,NPAGE,ISTATUS,INITSS,NUMDISP,IBEAD)          UCDATAB     2
C                                                                          UCDATAB     3
C     THIS ROUTINE UPDATES THE CURVE DISPLAY ACCESS TABLE.                 UCDATAB     4
C                                                                          UCDATAB     5
C                  *** THE CURVE DISPLAY ACCESS TABLE ***                  UCDATAB     6
C     THE CURVE DISPLAY ACCESS TABLE IS USED TO STORE ACCESS               UCDATAB     7
C     INFORMATION ABOUT THE CURVES AND/OR CURVE FITS WHICH BELONG TO       UCDATAB     8
C     THE PLOT ""WHICH IS DISPLAYED ON THE SCREEN"". THE INFORMATION       UCDATAB     9
C      ************************************                                UCDATAB    10
C     IN THE TABLE CAN BE USED TO:                                         UCDATAB    11
C          1 - DETERMINE IF A CURVE OR A CURVE FIT FOR A CURVE IN          UCDATAB    12
C              THE PLOT IS DISPLAYED ON THE SCREEN.                        UCDATAB    13
C          2 - ERASE A CURVE OR CURVE FIT IN THE PLOT, OR                  UCDATAB    14
C          3 - RETURN TO THE CURVE BEAD OR CURVE FIT BEAD(IF IT            UCDATAB    15
C              EXISTS) OF A CURVE IN THE PLOT.                             UCDATAB    16
C     THE I-TH ROW OF ICDATA3 IS INFORMATION ABOUT THE I-TH CURVE          UCDATAB    17
C     (CHRONOLOGICALLY) TO BE DISPLAYED ON THE SCREEN.                     UCDATAB    18
C                  ** PAGE ONE---CURVE INFORMATION **                      UCDATAB    19
C     ICDATAB(I,1,1) IS THE STATUS OF THE I-TH CURVE. IF                   UCDATAB    20
C                    ICDATAB(I,1,1) = 6HACTIVE, THEN THE CURVE IS          UCDATAB    21
C                    CURRENTLY DISPLAYED ON THE SCREEN. IF                 UCDATAB    22
C                    ICDATAB(I,1,1) = 8HINACTIVE, THEN THE CURVE HAS       UCDATAB    23
C                    BEEN DISPLAYED ON THE SCREEN AND ERASED.              UCDATAB    24
C     ICDATAB(I,2,1) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB    25
C                    CURVE. INITSS IS EXPLAINED IN SUBROUTINE              UCDATAB    26
C                    DISCRVE.                                              UCDATAB    27
C     ICDATAB(I,3,1) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB    28
C                    EXHIBIT THE I-TH CURVE.                               UCDATAB    29
C     ICDATAB(I,4,1) IS THE POINTER TO THE CURVE BEAD OF THE I-TH          UCDATAB    30
C                    CURVE.                                                UCDATAB    31
C                  ** PAGE TWO---CURVE FIT INFORMATION **                  UCDATAB    32
C     ICDATAB(I,1,2) IS THE STATUS OF THE I-TH CURVE FIT. IF               UCDATAB    33
C                    ICDATAB(I,1,2) = 4HNONE, THEN THE CURVE DOES          UCDATAB    34
C                    NOT HAVE A CURVE FIT. IF ICDATAB(I,1,2) =             UCDATAB    35
C                    6HACTIVE, THEN THE CURVE HAS A CURVE FIT AND IT       UCDATAB    36
C                    IS DISPLAYED ON THE SCREEN. IF ICDATAB(I,1,2) =       UCDATAB    37
C                    8HINACTIVE, THEN THE CURVE HAS A CURVE FIT BUT        UCDATAB    38
C                    IT IS NOT DISPLAYED ON THE SCREEN.                    UCDATAB    39
C     ICDATAB(I,2,2) IS THE INITIAL SUBSCRIPT (INITSS) FOR THE I-TH        UCDATAB    40
C                    CURVE FIT. INITSS IS EXPLAINED IN SUBROUTINE          UCDATAB    41
C                    DCRVFIT.                                              UCDATAB    42
C     ICDATAB(I,3,2) IS THE NUMBER OF DISPLAYS (NUMDISP) REQUIRED TO       UCDATAB    43
C                    EXHIBIT THE I-TH CURVE FIT.                           UCDATAB    44
C     ICDATAB(I,4,2) IS THE POINTER TO THE FIT BEAD OF THE I-TH            UCDATAB    45
C                    CURVE.                                                UCDATAB    46
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),    /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),            6JULY77     2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM   1SEPT       1
      ICDATAB(NROW,1,NPAGE) = ISTATUS                                      UCDATAB    48
      ICDATAB(NROW,2,NPAGE) = INITSS                                       UCDATAB    49
      ICDATAB(NROW,3,NPAGE) = NUMDISP                                      UCDATAB    50
      ICDATAB(NROW,4,NPAGE) = IBEAD                                        UCDATAB    51
      RETURN                                                               UCDATAB    52
      END                                                                  UCDATAB    53
```

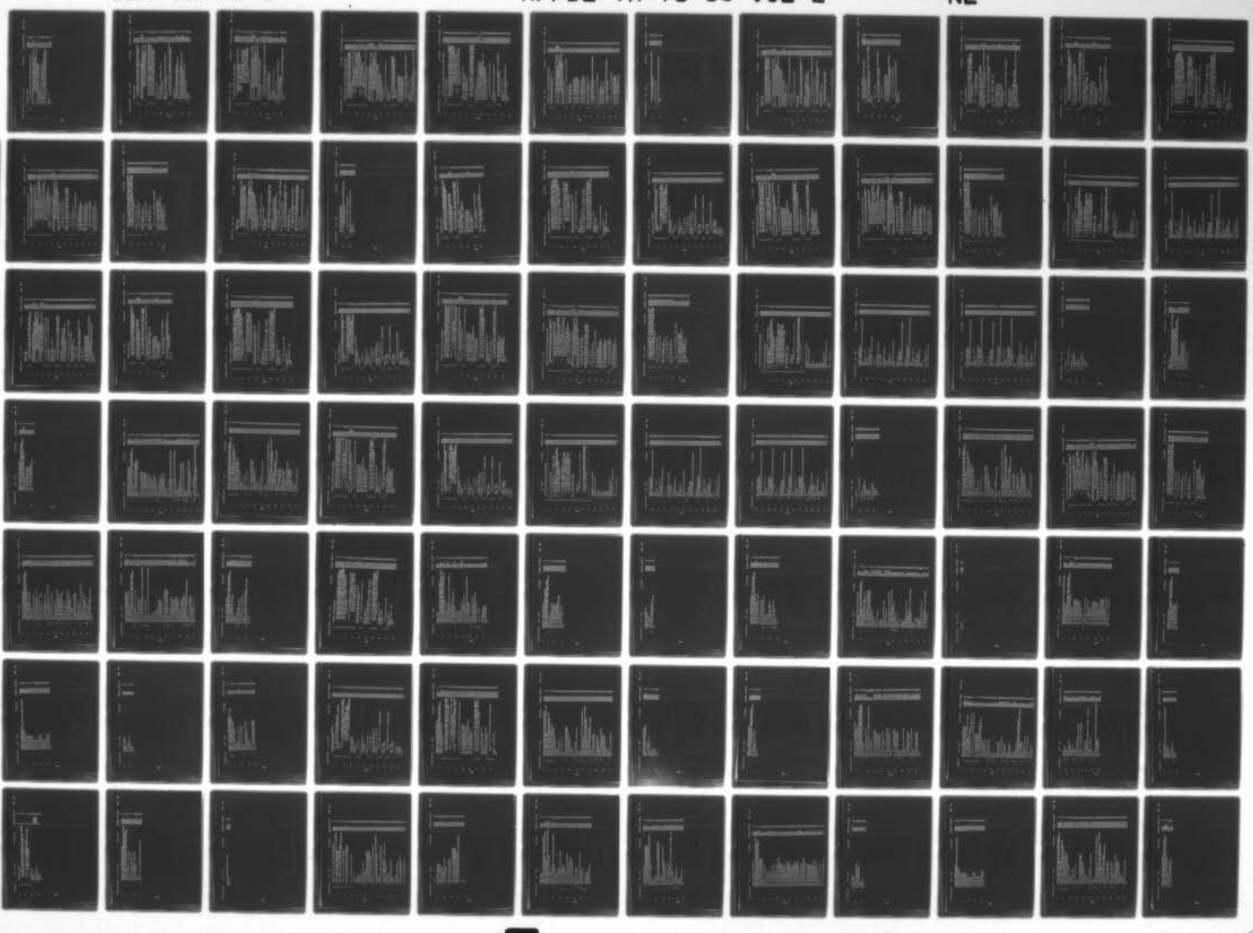
```
      OVERLAY (6,1)                                                     INHALE     69
      PROGRAM CONCF                                                     INHALE     70
C                                                                       CONCF       3
C         THIS ROUTINE GIVES THE USER THE OPPORTUNITY TO ACCEPT OR      CONCF       4
C         REJECT THE CURVE FITS WHICH ARE DISPLAYED ON THE SCREEN.      CONCF       5
C         THE DEFAULT DECISION IS ACCEPTANCE, I.E., IF THE USER         CONCF       6
C         ACCEPTS ALL OF THE CURVE FITS WHICH ARE DISPLAYED ON THE      CONCF       7
C         SCREEN, THEN HE MUST BUTTON PICK "STOP" OR HIS CHOICE FOR     CONCF       8
C         THE NEXT MENU HE WANTS TO SEE. HE DOES NOT HAVE TO SINGLE     CONCF       9
C         PICK OR STRING PICK ANY CURVE DESCRIPTIONS. IF THE USER       CONCF      10
C         REJECTS SOME OR ALL OF THE CURVE FITS WHICH ARE DISPLAYED     CONCF      11
C         ON THE SCREEN, THEN HE MUST MAKE HIS REJECTIONS KNOWN VIA     CONCF      12
C         A STRING PICK OR SINGLE PICK BEFORE BUTTON PICKING "STOP"     CONCF      13
C         OR HIS CHOICE FOR THE NEXT MENU.                              CONCF      14
C                                                                       CONCF      15
C         PRIMARY INPUT TO THE ROUTINE IS THE CURVE DISPLAY ACCESS      CONCF      16
C         TABLE. PRIMARY OUTPUT IS A SET OF ACCEPTABLE CURVE FITS.      CONCF      17
C                                                                       CONCF      18
      COMMON/BLOCK/IBLOCK(50)                                           /BLOCK/     2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/    2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/    3
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/    2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77     2
     Y               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT       1
      COMMON /GC/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBTTT      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77     3
      DIMENSION IDDTSTR(16), NCURVE(16)                                 CONCF      23
      EQUIVALENCE (IDDTSTR(1),IBLOCK(1)), (NCURVE(1),IBLOCK(17))        CONCF      24
      CALL CONINST                                                      CONCF      25
      CALL CONMENU                                                      CONCF      26
C                                                                       CONCF      27
C         WAIT FOR A BUTTON PICK FROM THE FOLLOWING MENU:               CONCF      28
C              81 - CURVE FIT MENU                                      CONCF      29
C              82 - MASTER CONTROL MENU                                 CONCF      30
C             254 - STOP                                                14APR76    64
C                                                                       CONCF      32
      CALL GICLR(NCON)                                                  CONCF      33
      CALL GIBUT(6,NCON,IDDTBUT,IDDCBUT)                                CONCF      34
      IF(IDDCBUT .EQ. 254) CALL STOP                                    14APR76    65
C                                                                       CONCF      36
C         ERASE THE CONFIRMATION INSTRUCTIONS, THE CONFIRMATION MENU,   CONCF      37
C         AND "ALL".                                                    8JUNE76    3
C              *                                                        CONCF      40
      CALL TERASE(IDINSTR,IDINSTR(19))                                  CONCF      41
      CALL TERASE(IDMENU,IDMENU(13))                                    CONCF      42
      CALL GIERAS(IDBLOCK(1))                                           CONCF      43
C                                                                       CONCF      84
C         REDISPLAY THE ROUTINES WHICH WERE ERASED IN                   CONCF      85
C         "CRVFIT" AND PROCESS THE BUTTON PICK.                         CONCF      86
C                                                                       CONCF      87
   40 CALL FRPLOT                                                       CONCF      88
C                                                                       CONCF      89
C         CHECK TO SEE IF "CURVE FIT MENU" WAS SELECTED.                CONCF      90
C                                                                       CONCF      91
      IF (IDDCBUT .EQ. 81) CALL AERTRN                                  FIXINH     23
C                                                                       CONCF      93
C         "MASTER CONTROL MENU" WAS SELECTED.                           CONCF      94
```

```
      C
            DO 50 J=1,NCIP                                                          CONCF      95
      60    IF (ICDATAB(J,1,1).EQ.6HACTIVE) CALL GIMOVE (-0,-0,-0,IDCO(J),IGN)      8JUNE76     4
         50 CONTINUE                                                                8JUNE76     5
            CALL STOPIT                                                             8JUNE76     6
            CALL ABTSKC(7HQTRBACK)                                                  DELRAD      9
            END                                                                     CONCF      96
                                                                                    CONCF      97
```

```
      SUBROUTINE CONINST                                                  CONINST    2
C                                                                         5MAY76    24
C     THIS ROUTINE DISPLAYS THE CURVE FIT CONFIRMATION INSTRUCTIONS.      5MAY76    25
C                                                                         5MAY76    26
      COMMON/DISPLAY/IDSTOP,IDDAD(81),NN,NNEND(10),IDSPEC(19),IDCF(41),   /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),           6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                    CYBTTT     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77    3
      CALL ORIGIN(-5.,-4.5)                                               5MAY76    29
      CALL SQUARE (0.,-1.75,4.7,1.75,IDINSTR(1),IGN,-0)                   8JUNE76    7
      CALL ANNOTA(1.53,.1,12HINSTRUCTIONS,12,IDINSTR(2),IGN,-0,-0,-0)     5MAY76    31
      CALL ANNOTA(.25,-.5,34H1.TO ACCEPT ALL OF THE CURVE FITS,,34,       5MAY76    32
     X            IDINSTR(3),IGN,-0,-0,-0)                              5MAY76    33
      CALL ANNOTA (.25,-.75,29H  PICK "MASTER CONTROL MENU".,             8JUNE76    8
     1     29,IDINSTR(4),IGN,-0,-0,-0)                                    8JUNE76    9
      CALL ANNOTA (.25,-1.25,34H2.TO REJECT ANY OR ALL CURVE FITS,,       8JUNE76   10
     1     34,IDINSTR(5),IGN,-0,-0,-0)                                    8JUNE76   11
      CALL ANNOTA (.25,-1.5,24H  PICK "CURVE FIT MENU".,24,               8JUNE76   12
     1     IDINSTR(6),IGN,-0,-0,-0)                                       8JUNE76   13
      IDINSTR(19)=6                                                       8JUNE76   14
      RETURN                                                              CONINST    3
      END                                                                 CONINST    4
```

AD-A058 673 DAYTON UNIV OHIO RESEARCH INST F/G 9/2
PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). VOLUME II--ETC(U)
MAY 78 J G JENSEN F33615-76-C-3076
UNCLASSIFIED UDR-TR-78-34 AFFDL-TR-78-55-VOL-2 NL

5 OF 7
AD A058 673



```
      SUBROUTINE CONMENU                                                    CONMENU    2
C                                                                           5MAY76    52
C         THIS SUBROUTINE DISPLAYS THE CURVE FIT CONFIRMATION MENU.         5MAY76    53
C                                                                           5MAY76    54
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),     /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),            6JULY77    2
     Y               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM   1SEPT      1
      COMMON /SP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                      CY6777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                    6JULY77    3
      CALL ORIGIN(1.2,-6.6)                                                 5MAY76    57
      CALL SQUARE(0.,-1.25,2.78,1.25,IDMENU(1),IGN,-0)                      5MAY76    58
      CALL ANNOTA(1.15,.1,-HMENU,4,IDMENU(2),IGN,-0,-0,-0)                  5MAY76    59
      CALL ANNOTA(.25,-.5,14HCURVE FIT MENU,14,IDMENU(3),IBUT,81,-0,-0)     5MAY76    60
      CALL ANNOTA(.25,-1.,19HMASTER CONTROL MENU,19,IDMENU(4),IBUT,82,      5MAY76    61
     X            -0,-0)                                                   5MAY76    62
      IDMENU(13) = 4                                                        5MAY76    63
      RETURN                                                                CONMENU    3
      END                                                                   CONMENU    4
```

```
      SUBROUTINE REPLOT                                                  REPLOT      2
C                                                                        REPLOT      3
C         THIS ROUTINE SEARCHES THROUGH THE CURVE DISPLAY ACCESS TABLE   REPLOT      4
C         AND REDISPLAYS ALL CURVES LISTED "INACTIVE" IN THE TABLE.      REPLOT      5
C                                                                        REPLOT      6
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/   2
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/   2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77     2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIFM 1SEPT       1
C                                                                        REPLOT      9
C         LOOP THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE. IF A  REPLOT     10
C         CURVE IS ACTIVE, INCREASE THE IDDAD INDEX(NN) BY THE NUMBER OF REPLOT     11
C         ELEMENTS OF IDDAD REQUIRED TO DISPLAY IT AND THEN GO TO THE    REPLOT     12
C         NEXT ONE. IF A CURVE IS INACTIVE, THEN DISPLAY IT AND ITS      REPLOT     13
C         DESCRIPTION AND UPDATE THE CURVE DISPLAY ACCESS TABLE.         REPLOT     14
C                                                                        REPLOT     15
      CALL ORIGIN(-5.,-3.)                                               5MAY76      7
      NN = 0                                                             REPLOT     16
      DO 100 I=1,NCIP                                                    3JUNE76     1
      IF(ICDATAB(I,1,1) .EQ. 8HINACTIVE) GO TO 10                        14APR76     5
      NN = NN + ICDATAB(I,3)                                             REPLOT     19
      GO TO 30                                                           REPLOT     20
   10 CALL DISCRVE (ICDATAB(I,4,1),ISYMBOL,INITSS,NUMDISP,I)             ADPTPRE     7
      IF(NUMDISP .GT. 0) GO TO 20                                        REPLOT     22
C                                                                        REPLOT     23
C         THIS CODE IS HERE TO COVER THE POSSIBILITY THAT EVERY POINT ON REPLOT     24
C         A CURVE HAS BEEN DELETED. THE CURVE IS ACTIVE BUT EMPTY.       REPLOT
C                                                                        REPLOT
      NN = NN + 1                                                        REPLOT     27
      IDDAD(NN) = 0                                                      REPLOT     28
      CALL DCRVDES (ICDATAB(I,4,1),ISYMBOL,I)                            25MAY76     4
      CALL UCDATAB(I,1,6HACTIVE,INITSS,1,ICDATAB(I,4,1))                 14APR76     6
      GO TO 30                                                           REPLOT     31
   20 CALL DCRVDES (ICDATAB(I,4,1),ISYMBOL,I)                            25MAY76     5
      CALL UCDATAB(I,1,6HACTIVE,INITSS,NUMDISP,ICDATAB(I,4,1))           14APR76     7
   30 IF (ICDATAB(I,4,2).LE.0) GO TO 100                                 3JUNE76     2
      IF (ICDATAB(I,1,2).EQ.6HACTIVE) GO TO 100                          3JUNE76     3
      CALL REPLOF (I,I1,I2,1)                                            20SEPT     23
  100 CONTINUE                                                           3JUNE76     5
      RETURN                                                             REPLOT     35
      END                                                                REPLOT     36
```

```
      SUBROUTINE DCRVDES(ICRVPTR,ISYMBOL,NCURVE)                        DCRVDES    2
C                                                                       DCRVDES    3
C         THIS ROUTINE BUILDS A CURVE DESCRIPTION AND DISPLAYS IT.      DCRVDES    4
C              ICRVPTR - THE CURVE BEAD POINTER.                        DCRVDES    5
C              ISYMBOL - THE CURVE PLOT SYMBOL.                         DCRVDES    6
C               NCURVE - THE NUMBER OF THE CURVE. THE CURVES ARE        DCRVDES    7
C                        NUMBERED CHRONOLOGICALLY AS THEY APPEAR        DCRVDES    8
C                        ON THE SCREEN. THAT NUMBER IS USED IN          DCRVDES    9
C                        THIS SUBROUTINE TO DETERMINE THE POSITION      DCRVDES   10
C                        ON THE SCREEN WHERE THE CURVE DESCRIPTION      DCRVDES   11
C                        WILL BE DISPLAYED.                             DCRVDES   12
C                                                                       DCRVDES   13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                 /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTF(19),IDDATAR(20,4,2),NCIP,IDBLOCK(20),NCIPN 1SEPT      1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBER77    1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      INTEGER RECORD(3)                                                 DCRVDES   17
C                                                                       DCRVDES   18
C         GET THE PARAMETER NAME AND VALUE FROM THE CURVE BEAD, ENCODE  DCRVDES   19
C         THE CURVE DESCRIPTION, AND DISPLAY IT.                        DCRVDES   20
C                                                                       DCRVDES   21
      CALL DMGET(ICPARAN,ICRVPTR,NAME)                                  DCRVDES   22
      CALL DMGET(ICPARAV,ICRVPTR,VALUE)                                 DCRVDES   23
      IDEC=8                                                            FIXINH     2
      DO 10 J=1,9                                                       FIXINH     3
      IF (VALUE/10.**(J-1).LE.1.) GO TO 20                              FIXINH     4
   10 IDEC=IDEC-1                                                       FIXINH     5
      IDEC=1                                                            FIXINH     6
   20 ENCODE (25,1000,RECORD) ISYMBOL,NAME,IDEC,VALUE                   FIXINH     7
 1000 FORMAT(I1,1X,A10,1X,F12.=)                                        FIXINH     8
      Y = 3.0 - .5 * FLOAT(NCURVE)                                      DCRVDES   26
      CALL ANNOTA (10.43,Y,RECORD,25,IDCD(NCURVE),IGN,NCURVE,-0,-0)     6JULY77    6
      RETURN                                                            DCRVDES   28
      END                                                               DCRVDES   29
```

```
      SUBROUTINE DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NPLOT)                 ADPTPRE    1
C                                                                               DISCRVE    3
C     THIS ROUTINE DISPLAYS THE CURVE(SET OF POINTS) INDICATED BY               DISCRVE    4
C     "ICRVPTR". THE PRIMARY INPUTS TO THE ROUTINE ARE THE POINTER              DISCRVE    5
C     TO THE CURVE BEAD AND THE PLOT SCALE DATA. PRIMARY OUTPUTS                DISCRVE    6
C     FROM THE ROUTINE ARE:                                                     DISCRVE    7
C          ISYMBOL - THE SYMBOL USED TO DISTINGUISH THIS CURVE FROM             DISCRVE    8
C                    OTHER CURVES ON THE PLOT.                                  DISCRVE    9
C           INITSS - THE DISPLAY ADDRESSES FOR THE CURVE ARE STORED             DISCRVE   10
C                    IN ARRAY IODAD. IODAD(INITSS) CONTAINS THE                 DISCRVE   11
C                    DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED              DISCRVE   12
C                    TO EXHIBIT THE CURVE.(INITSS-INITIAL SUBSCRIPT)            DISCRVE   13
C          NUMDISP - NUMDISP CONTAINS THE NUMBER OF DISPLAYS                    DISCRVE   14
C                    REQUIRED TO EXHIBIT THE CURVE.                             DISCRVE   15
C          NPLOT   - NPLOT CONTAINS THE NUMBER OF THE                           ADPTPRE    2
C                    PLOT IN THE TABLE                                          ADPTPRE    3
C     INITSS AND NUMDISP ARE NEEDED TO ERASE THE CURVE. THEY WILL BE            DISCRVE   16
C     STORED IN THE CURVE DISPLAY ACCESS TABLE.                                 DISCRVE   17
C                                                                               DISCRVE   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                            /AXSCALE/    2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,          /CRVBEAD/    2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                         /CRVBEAD/    3
      COMMON/DISPLAY/IOSTOP,IODAD(80),NN,NNEND(10),IDSPEC(19),IOCF(41),         /DISPLAY/    2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),                  6JULY77      2
     X               IOINSTR(19),IODATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM         1SEPT        1
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                          CYB777       1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                         6JULY77      3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                        /PARMSV/     2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                     /POINTS/     2
      COMMON /SYMTAB/ ISYMTAB(20)                                               /SYMTAB/     2
      DIMENSION MESG(7)                                                         9JULY76      1
C                                                                               DISCRVE   28
C     RECORD THE INITIAL VALUE OF THE IODAD SUBSCRIPT.                          DISCRVE   29
C                                                                               DISCRVE   30
      INITSS=(NPLOT-1)*4+1                                                      ADPTPRE    4
      NN=INITSS-1                                                               ADPTPRE    5
C                                                                               DISCRVE   32
C     GET THE NUMBER OF POINTS ON THE CURVE, THE CURVE SYMBOL                   DISCRVE   33
C     NUMBER, THE RECORD BEAD POINTER, THE X-COORDINATE BEAD                    DISCRVE   34
C     POINTER, AND THE Y-COORDINATE BEAD POINTER FROM THE                       DISCRVE   35
C     CURVE BEAD.                                                               DISCRVE   36
C                                                                               DISCRVE   37
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                        DISCRVE   38
      CALL DMGET(ICSYM,ICRVPTR,ISYMBOL)                                         DISCRVE   39
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                         DISCRVE   40
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                          DISCRVE   41
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                          DISCRVE   42
C                                                                               DISCRVE   43
C     IF THE NUMBER OF POINTS EXCEEDS THE PROGRAMMED LIMIT(248),                DISCRVE   44
C     DISPLAY THE FIRST 248 POINTS ONLY.                                        DISCRVE   45
C                                                                               DISCRVE   46
      IF(NPOINTS .LE. 248) GO TO 10                                             DISCRVE   47
      NPOINTS=248                                                               DISCRVE   48
      CALL DMSET(ICNPTS,ICRVPTR,NPOINTS)                                        DISCRVE   49
      CALL ERROFM(0.,7.,62HPOINTS LIMIT EXCEEDED. THE FIRST 248 POINTS W        DISCRVE   50
     X,10,62,7HDISCRVE,6HINFORM)                                                DISCRVE   51
C                                                                               DISCRVE   52
```

```
      C         CONVERT THE CURVE SYMBOL NUMBER INTO A CURVE SYMBOL.                DISCRVE   53
      C                                                                             DISCRVE   54
   60    10 ISYMBOL = ISYMTAB(ISYMBOL)                                              DISCRVE   55
      C                                                                             DISCRVE   56
      C         MOVE THE CONTENTS OF THE RECORD NUMBER BEAD, THE X-COORDINATE       DISCRVE   57
      C         BEAD AND THE Y-COORDINATE BEAD INTO CORE. THE RECORD NUMBER         DISCRVE   58
      C         BEAD IS A STATUS BEAD FOR THE POINTS IN THE OTHER TWO BEADS.        DISCRVE   59
   65 C              ACTIVE - THE RECORD NUMBER IS POSITIVE OR SHADDED. THE         DISCRVE   60
      C                       RECORD NUMBER IS SHADDED IF AND ONLY IF THE           DISCRVE   61
      C                       POINT HAS BEEN ADDED DURING AN EDIT.                  DISCRVE   62
      C            INACTIVE - THE RECORD NUMBER IS NEGATIVE. A POINT IS           DISCRVE   63
      C                       INACTIVE IF AND ONLY IF IT HAS BEEN DELETED          DISCRVE   64
   70 C                       DURING AN EDIT.                                      DISCRVE   65
      C                                                                             DISCRVE   66
          KOMPCOD = 700000000010 + NPOINTS * 8**6                                   DISCRVE   67
          CALL DMGET(KOMPCOD,IRECPTR,NREC)                                          DISCRVE   68
          CALL DMGET(KOMPCOD,IXPTR,X)                                               DISCRVE   69
   75     CALL DMGET(KOMPCOD,IYPTR,Y)                                               DISCRVE   70
      C                                                                             DISCRVE   71
      C         LOOP THROUGH THE POINTS. IF A POINT IS INACTIVE OR NOT ON THE       DISCRVE   72
      C         PLOT, SKIP IT. OTHERWISE, CONVERT THE COORDINATES TO DGUS AND       DISCRVE   73
      C         PACK THE DESCRIPTION OF THE POINT INTO THE DESCRIPTION BUFFER.      DISCRVE   74
   80 C                                                                             DISCRVE   75
      12  DO 20 I=1,NPOINTS                                                        9JULY76    2
          IF(NREC(I) .LT. 0) GO TO 20                                               DISCRVE   77
          IF ((X(I).LT.XMIN).OR.(X(I).GT.XMAX))GO TO 20                          30JULY76    1
          IF ((Y(I).LT.YMIN).OR.(Y(I).GT.YMAX))GO TO 20                          30JULY76
   85     IX=IFIX(((X(I)-XMIN)*DGUPIN)/XINCR+XORDGU-6)                           5OCT
          IY=IFIX(((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU - 11.5)                   DISCRVE   81
          CALL GUPSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                                 DISCRVE   82
          CALL GUAN(ISYMBOL,1,IBUF,NBYTE,MBYTE)                                     DISCRVE   83
          IF(NBYTE .LT. 312) GO TO 20                                               DISCRVE   84
   90 C                                                                             DISCRVE   85
      C         THE DESCRIPTION BUFFER IS FULL. FLUSH IT.                           DISCRVE   86
      C                                                                             DISCRVE   87
          NN=NN+1                                                                   DISCRVE   88
          CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                                DISCRVE   89
   95     20 CONTINUE                                                               DISCRVE   90
      C                                                                             DISCRVE   91
      C         ALL POINTS HAVE BEEN CONSIDERED. IS THE DESCRIPTION                 DISCRVE   92
      C         BUFFER EMPTY? IF NOT, FLUSH IT.                                     DISCRVE   93
      C                                                                             DISCRVE   94
  100     IF(NBYTE .EQ. 0) GO TO 30                                                 DISCRVE   95
          NN=NN+1                                                                   DISCRVE   96
          CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(NN),IGN)                                DISCRVE   97
          30 NUMDISP = NN - INITSS + 1                                              DISCRVE   98
          RETURN                                                                    DISCRVE   99
  105     END                                                                       DISCRVE  100
```

```
      SUBROUTINE REPLCF (NCURVE,ICRVPTR,IFITPTR,IFLAG)                  20SEPT     8
C                                                                       REPLCF     3
C     THIS ROUTINE REPLOTS THE CURVE FITS OF THE                        REPLCF     4
C     CURVES WHICH ARE MARKED "INACTIVE".                               REPLCF     5
C                                                                       REPLCF     6
C                                                                       REPLCF     7
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDFP,ICFIT,ICSTAT                 /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPN 1SEPT      1
      COMMON /FITBEAD/ IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA             6REPLCF   10
      DIMENSION X(248),Y(248),FP(248)                                   REPLCF    11
C                                                                       REPLCF    12
C     BUFFY THE CURVE AND CURVE FIT BEADS FOR THE CALL TO DCRVFIT       REPLCF    13
C                                                                       REPLCF    14
      IF (IFLAG.LE.0) GO TO 20                                          20SEPT     9
      IFITPTR=ICDATAB(NCURVE,4,2)                                       REPLCF    15
      ICRVPTR=ICDATAB(NCURVE,4,1)                                       REPLCF    16
C                                                                       REPLCF    17
   20 CALL CMGET (IFTYPE,IFITPTR,ITYPE)                                 13SEPT     7
      CALL CMGET (ICINDFP,ICRVPTR,IVAR)                                 REPLCF    19
      CALL CMGET (IFIVVFP,IFITPTR,XINDFP)                               REPLCF    20
      CALL CMGET (IFIVVLP,IFITPTR,XINDLP)                               REPLCF    21
      CALL CMGET (IFIVI,IFITPTR,XINC)                                   REPLCF    22
C                                                                       REPLCF    23
      IF (ITYPE.EQ.6HAITKEN) GO TO 60                                   20SEPT    10
      CALL CMGET (IFNPARA,IFITPTR,NFTPA)                                REPLCF    24
      KOMPCOD=7000000000.6B + SHIFT(NFTPA,18)                           REPLCF    25
      CALL CMGET (KOMPCOD,IFITPTR,FP)                                   REPLCF    26
C                                                                       REPLCF    27
C     IF FIT IS LEAST SQUARES CALL DCRVFIT                              REPLCF    28
C     SPLINE NEEDS MORE INFORMATION                                     REPLCF    29
C                                                                       REPLCF    30
      IF (ITYPE.EQ.6HSPLINE) GO TO 50                                   REPLCF    31
      CALL DCRVFIT (ITYPE,IVAR,XDUM,YDUM,NDUM,FP,NFTPA,XINDFP,XINDLP,   REPLCF    32
     1    XINC,INITSS,NUMDISP,NCURVE)                                   REPLCF    33
      GO TO 100                                                         REPLCF    34
C                                                                       REPLCF    35
C     SPLINE FIT -- RETRIEVE EXTRA INFORMATION                          REPLCF    36
C                                                                       REPLCF    37
   50 KOMPCOD=KOMPCOD+NFTPA                                             REPLCF    38
      CALL CMGET (KOMPCOD,IFITPTR,X)                                    REPLCF    39
      KOMPCOD=KOMPCOD+NFTPA                                             REPLCF    40
      CALL CMGET (KOMPCOD,IFITPTR,Y)                                    REPLCF    41
C                                                                       REPLCF    42
      CALL DCRVFIT (ITYPE,IVAR,X,Y,NFTPA,FP,NFTPA,XINDFP,XINDLP,        REPLCF    43
     1    XINC,INITSS,NUMDISP,NCURVE)                                   REPLCF    44
      GO TO 100                                                         20SEPT    11
C                                                                       20SEPT    12
C         AITKEN FIT -- RETRIEVE X AND Y ARRAYS                         20SEPT    13
C                                                                       20SEPT    14
   60 CALL CMGET (IFNPARA,IFITPTR,NPOINTS)                              20SEPT    15
      CALL CMGET (IFNPARA+1,IFITPTR,NFTPA)                              5OCT       5
      KOMPCOD=7000000000B+SHIFT(NPOINTS,18)                             20SEPT    17
      CALL CMGET (KOMPCOD,IFITPTR,X)                                    20SEPT    18
      CALL CMGET (KOMPCOD+NPOINTS,IFITPTR,Y)                            20SEPT    19
```

```
      C                                                                  20SEPT   20
            CALL DCRVFIT (ITYPE,IVAR,X,Y,NPOINTS,FP,NFTPA,XINDFP,XINDLP,  20SEPT   21
     60   1     XINC,INITSS,NUMDISP,NCURVE)                              20SEPT   22
      C                                                                  REPLCF   45
      C     UPDATE CURVE DISPLAY TABLE                                   REPLCF   46
      C                                                                  REPLCF   47
        10  CALL UCDATAB (NCURVE,2,6HACTIVE,INITSS,NUMDISP,IFITPTR)      REPLCF   48
     65     RETURN                                                       REPLCF   49
            END                                                          REPLCF   50
```

```
      OVERLAY (6,2)                                                     FIXINH    24
      PROGRAM LEASTSQ                                                   FIXINH    25
C                                                                       LEASTSQ    3
C         THIS ROUTINE SUPERVISES THE CONSTRUCTION, STORAGE, AND DISPLAY LEASTSQ    4
C         OF LEAST SQUARES FITS. PRIMARY INPUT TO THE ROUTINE IS THE    LEASTSQ    5
C         CURVE DISPLAY ACCESS TABLE. PRIMARY OUTPUT IS FIT BEADS AND   LEASTSQ    6
C         DISPLAYED CURVE FITS FOR EACH OF THE ACTIVE CURVES LISTED IN  LEASTSQ    7
C         THE CURVE DISPLAY ACCESS TABLE.                               LEASTSQ    8
C                                                                       LEASTSQ    9
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,  /CRVBEAD/  2
     X             ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                   /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NMEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77    2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT      1
      COMMON/FITBEAD/IFTYPE,IFIVVEP,IFIVVLP,IFIVI,IFNPARA               14APR76    1
      COMMON /GE/ IBUF(126),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY0777     1
     X     ISTR,IPUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      DIMENSION WORK(42),W(248),POLYCF(7)                               18JUNE76  14
      CALL ORIGIN(-5.,-3.)                                              LEASTSQ   18
      CALL GIEOM (NCON,IEOM,IBUT,253)                                   CY0777     9
C                                                                       LEASTSQ   20
C         GET THE DEGREE OF THE LEAST SQUARES POLYNOMIAL FROM THE USER. LEASTSQ   21
C                                                                       LEASTSQ   22
    5 CALL GETDEG (NDEG)                                                3MARCH    32
C                                                                       LEASTSQ   24
C         GET THE INDEPENDENT VARIABLE FROM THE USER.                   LEASTSQ
C                                                                       LEASTSQ
      CALL GETIV(IV)                                                    LEASTSQ   27
      CALL GIEOM(NCON)                                                  LEASTSQ   28
C                                                                       LEASTSQ   29
C         LOOP THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE. IF A LEASTSQ   30
C         CURVE IS INACTIVE, SKIP IT. OTHERWISE, CURVE FIT IT.          LEASTSQ   31
C                                                                       LEASTSQ   32
      DO 20 I=1,NCIP                                                    LEASTSQ   33
      IF(ICDATAB(I,1,1) .EQ. 8HINACTIVE) GO TO 20                       LEASTSQ   34
      IF(ICDATAB(I,1,2) .EQ. 8HINACTIVE) GO TO 10                       LEASTSQ   35
      CALL TERASE(ICCF(ICDATAB(I,2,2)),ICDATAB(I,3,2))                  14APR76   77
      CALL CHRL3D(ICDATAB(I,4,2))                                       LEASTSQ   37
C                                                                       LEASTSQ   38
C         MOVE THE POINTS ARRAYS INTO CORE AND, IF NECESSARY, SORT THEM. LEASTSQ   39
C                                                                       LEASTSQ   40
   10 CALL SORTCK(ICDATAB(I,4,1),IV,NPOINTS)                            LEASTSQ   41
C                                                                       LEASTSQ   42
C         CLEAN UP THE IN-CORE POINTS ARRAYS BY REMOVING ALL POINTS     LEASTSQ   43
C         FROM THEM EXCEPT THOSE DISPLAYED ON THE SCREEN.               LEASTSQ   44
C                                                                       LEASTSQ   45
      CALL HRCLEAN(NPOINTS)                                             LEASTSQ   46
      IF (NPOINTS.LE.1) CALL ABORTN                                     INHALE    77
C                                                                       LEASTSQ   47
C         GET A LEAST SQUARES FIT. THE LEAST SQUARES                    LEASTSQ   48
C         COEFFICIENTS ARE RETURNED IN ARRAY "POLYCF".                  LEASTSQ   49
C                                                                       LEASTSQ   50
      NCOEFF = NDEG + 1                                                 LEASTSQ   51
      W(1)=-1.                                                          18JUNE76  15
      I=0                                                               18JUNE76  16
      IF (IV.EQ.1HX) CALL PLSCF(X,Y,W,NPOINTS,NDEG,NMAX,                18JUNE76  17
```

```
                1     POLYCF,IN,XD,XO,WORK,IER)                                  18JUNE76   18
                  IF (IV.EQ.1HY) CALL PLSCF (Y,X,W,NPOINTS,NDEG,NMAX,            18JUNE76   19
60              1     POLYCF,IN,XD,XO,WORK,IER)                                  18JUNE76   20
                  IF (IER.EQ.0) GO TO 15                                         3MARCH     33
                  CALL ERROAM(1.,3.5,54HERRORS IN LEAST SQUARES--POSSIBLE BAD INDEPE LASTBUX    7
                 6NDENT VARIABLE,56,7HLEASTSQ,6HINFORM)                          LASTBUX    8
                  GO TO 5                                                        3MARCH     36
65              15 CONTINUE                                                      3MARCH     37
            C                                                                    LEASTSQ    59
            C         BUILD A FIT BEAD.                                          LEASTSQ    60
            C                                                                    LEASTSQ    61
                  CALL BUILDFB(10HLEASTSQUAR,POLYCF,NCOEFF,IV,NPOINTS,IFITPTR)  LEASTSQ    62
70                CALL DMSET(ICFIT,ICDATAB(I,4,1),IFITPTR)                       LEASTSQ    63
            C                                                                    LEASTSQ    64
            C         DISPLAY THE FIT.                                           LEASTSQ    65
            C                                                                    LEASTSQ    66
                  CALL DMGET(IFIVVFP,IFITPTR,TMIN)                               LEASTSQ    67
75                CALL DMGET(IFIVVLP,IFITPTR,TMAX)                               LEASTSQ    68
                  CALL DMGET(IFIVI,IFITPTR,TINCR)                                LEASTSQ    69
                  CALL DCRVFIT(10HLEASTSQUAR,IV,X,Y,NPOINTS,POLYCF,NCOEFF,       LEASTSQ    70
                 X          TMIN,TMAX,TINCR,INITSS,NUMDISP,I)                    28MAY76    3
            C                                                                    LEASTSQ    72
80          C         UPDATE THE CURVE DISPLAY ACCESS TABLE.                     LEASTSQ    73
            C                                                                    LEASTSQ    74
                  CALL UCDATAB(I,2,6HACTIVE,INITSS,NUMDISP,IFITPTR)              LEASTSQ    75
               20 CONTINUE                                                       LEASTSQ    76
                  CALL AERTRN                                                    INHALE     78
85                END                                                            LEASTSQ    78
```

```
      SUBROUTINE GETDEG(NDEG)                                                  GETDEG      2
C                                                                              GETDEG      3
C         THIS ROUTINE SOLICITS THE DEGREE OF THE LEAST SQUARES                GETDEG      4
C         POLYNOMIAL FROM THE USER, CHECKS TO SEE THAT THE SUBMITTED           GETDEG      5
C         VALUE IS IN RANGE (1 - 10), AND RETURNS AN ADMISSABLE VALUE          GETDEG      6
C         TO THE CALLING PROGRAM.                                              GETDEG      7
C                                                                              GETDEG      8
C         NDEG - 0 - THE SUBMITTED DEGREE OF THE                               GETDEG      9
C                    LEAST SQUARES POLYNOMIAL.                                 GETDEG     10
C                                                                              GETDEG     11
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                         CY8777      1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                         6JUL77      3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                       /PARMSV/    2
   10 CALL ANNOTA(0.12,-3.0,63HENTER THE DEGREE OF THE LEAST SQUARES POL       GETDEG     14
     XYNOMIAL (1 TO 6) ---  ,63,IDMESG,IGN,-0,-0,-0)                          18JUNE76    12
C                                                                              GETDEG     16
C         DISPLAY THE PROMPTING MESSAGE AND LIGHT REGISTER, RETRIEVE           GETDEG     17
C         THE DEGREE, AND ERASE THE PROMPTING MESSAGE.                         GETDEG     18
C                                                                              GETDEG     19
      IX = IFIX(7.68 * DGUPIN + XORDGU + .5)                                   GETDEG     20
      IY = IFIX(-3.0 * DGUPIN + YORDGU + .5)                                   GETDEG     21
      NC = 10                                                                  GETDEG     22
      CALL GIANS(NCON,NC,IX,IY)                                                GETDEG     23
   20 CALL GIBUT(0,NCON,IDOT,IDDC)                                             GETDEG     24
      IF(IDDC .EQ. 254) CALL STOP                                              14APR76    72
      IF(IDDC .NE. 253) GO TO 20                                               14APR76    73
      CALL GIANE(NCON,NC,NDEGDC)                                               GETDEG     27
      CALL GIERAS(IDMESG)                                                      GETDEG     28
C                                                                              GETDEG     30
C         CHECK TO SEE THAT THE SUBMITTED DEGREE IS ADMISSABLE.                GETDEG     31
C                                                                              GETDEG     32
      IF(NC .LT. 1) GO TO 30                                                   GETDEG     33
      ENCODE(4,1000,IFMT) NC                                                   GETDEG     34
 1000 FORMAT(2H(I,I1,1H))                                                      GETDEG     35
      DECODE(NC,IFMT,NDEGDC) NDEG                                              GETDEG     36
      IF ( (NDEG.GE.1) .AND. (NDEG.LE.6) ) RETURN                              18JUNE76    13
   30 CALL ERROR(1.2,-3.0,47HTHE DEGREE WHICH YOU SUBMITTED IS OUT OF R        GETDEG     38
     XANGE.,47,6HGETDEG,6HINFORM)                                             GETDEG     39
      GO TO 10                                                                 GETDEG     40
      END                                                                      GETDEG     41
```

```
      SUBROUTINE GETIV(IV)                                              GETIV      2
C                                                                       GETIV      3
C         THIS ROUTINE SOLICITS THE INDEPENDENT VARIABLE FROM THE USER, GETIV      4
C         CHECKS TO SEE THAT THE SUBMITTED INDEPENDENT VARIABLE IS      GETIV      5
C         ACCEPTABLE, AND RETURNS IT TO THE CALLING PROGRAM.            GETIV      6
C                                                                       GETIV      7
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYB777     1
     X    ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                /PARMSV/   2
C                                                                       GETIV     10
C         DISPLAY THE PROMPTING MESSAGE AND LIGHT REGISTER, RETRIEVE    GETIV     11
C         THE INDEPENDENT VARIABLE, AND ERASE THE PROMPTING MESSAGE.    GETIV     12
C                                                                       GETIV     13
   10 CALL ANNOTA(1.86,-3.0,44HENTER THE INDEPENDENT VARIABLE (X OR Y) - GETIV     14
     X-- ,44,IDMESG,IGN,-0,-0,-0)                                       GETIV     15
      IX = IFIX(7.14 * DGUPIN + XORDGU + .5)                            GETIV     16
      IY = IFIX(-3.0 * DGUPIN + YORDGU + .5)                            GETIV     17
      NC=2                                                              CYB777     8
      CALL GIANS(NCON,NC,IX,IY)                                         GETIV     19
   20 CALL GIBUT(0,NCON,IDOT,IDOC)                                      GETIV     20
      IF(IDOC .EQ. 254) CALL STOP                                       14APR76   74
      IF(IDOC .NE. 253) GO TO 20                                        14APR76   75
      CALL GIANE(NCON,NC,IV)                                            GETIV     23
      CALL GIERAS(IDMESG)                                               GETIV     24
C                                                                       GETIV     25
C         CHECK TO SEE THAT THE SUBMITTED INDEPENDENT                   GETIV     26
C         VARIABLE IS ADMISSABLE.                                       GETIV
C                                                                       GETIV
      IF( (IV .EQ. 1HX) .OR. (IV .EQ. 1HY) ) RETURN                     GETIV     29
      CALL ERRORM(1.0,-3.0,29HYOU DID NOT INPUT "X" OR "Y".,29,         GETIV     30
     X            5HGETIV,6HINFORM)                                    GETIV     31
      GO TO 10                                                          GETIV     32
      END                                                               GETIV     33
```

```
      SUBROUTINE MRCLEAN(NPOINTS)                                        MRCLEAN     2
C                                                                        MRCLEAN     3
C         THIS ROUTINE CLEANS UP THE CORE RESIDENT POINTS ARRAYS BY      MRCLEAN     4
C         REMOVING INACTIVE POINTS AND POINTS WHICH ARE OFF SCALE FROM   MRCLEAN     5
C         THE IN-CORE POINTS ARRAYS. PRIMARY INPUTS TO THE ROUTINE ARE   MRCLEAN     6
C         THE POINTS ARRAYS, THE STATUS ARRAY, AND THE SCALE INFORMATION MRCLEAN     7
C         THE PRIMARY OUTPUT FROM THE ROUTINE IS SANITIZED POINTS ARRAYS MRCLEAN     8
C         WHICH ARE READY TO BE FED TO THE CURVE FITTING ROUTINES.       MRCLEAN     9
C         NOTE---THE DATA MANAGER FILE CONTINUES TO HOLD THE             MRCLEAN    10
C               UNSANITIZED ARRAYS.                                      MRCLEAN    11
C                                                                        MRCLEAN    12
C         NPOINTS - I/O - WHEN NPOINTS IS INPUT, IT IS EQUAL TO THE      MRCLEAN    13
C                   NUMBER OF POINTS IN THE COORDINATE READS. WHEN       MRCLEAN    14
C                   NPOINTS IS OUTPUT, IT IS EQUAL TO THE NUMBER OF      MRCLEAN    15
C                   POINTS WHICH ARE DISPLAYED ON THE SCREEN. THE        MRCLEAN    16
C                   TWO ARE NOT NECESSARILY THE SAME.                    MRCLEAN    17
C                                                                        MRCLEAN    18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/    2
C                                                                        MRCLEAN    21
C         NPNOP --- THE NUMBER OF POINTS NOT ON THE PLOT.                MRCLEAN    22
C                                                                        MRCLEAN    23
      NPNOP = 0                                                          MRCLEAN    24
      DO 20 I=1,NPOINTS                                                  MRCLEAN    25
      IF(NREC(I) .LT. 0) GO TO 10                                        MRCLEAN    26
      IF( (X(I) .LT. XMIN) .OR. (X(I) .GT. XMAX) ) GO TO 10              MRCLEAN    27
      IF( (Y(I) .LT. YMIN) .OR. (Y(I) .GT. YMAX) ) GO TO 10              MRCLEAN    28
C                                                                        MRCLEAN    29
C         THE "I-TH" POINT IS ON THE PLOT(CURRENTLY DISPLAYED ON THE     MRCLEAN    30
C         SCREEN). IF ANY POINTS PRIOR TO THE "I-TH" POINT ARE NOT ON    MRCLEAN    31
C         THE PLOT, THEN MOVE THE "I-TH" POINT INTO THE GAP CREATED      MRCLEAN    32
C         BY THE MISSING POINT(S). OTHERWISE, GO TO THE "(I+1)-ST" POINT MRCLEAN    33
C                                                                        MRCLEAN    34
      IF(NPNOP .EQ. 0) GO TO 20                                          MRCLEAN    35
      X(I-NPNOP) = X(I)                                                  MRCLEAN    36
      Y(I-NPNOP) = Y(I)                                                  MRCLEAN    37
      GO TO 20                                                           MRCLEAN    38
C                                                                        MRCLEAN    39
C         THE POINT IS NOT ON THE PLOT.                                  MRCLEAN    40
C                                                                        MRCLEAN    41
   10 NPNOP = NPNOP + 1                                                  MRCLEAN    42
   20 CONTINUE                                                           MRCLEAN    43
      NPOINTS = NPOINTS - NPNOP                                          MRCLEAN    44
      RETURN                                                             MRCLEAN    45
      END                                                                MRCLEAN    46
```

```
      SUBROUTINE SORTER(NPOINTS,IV)                                     SORTER     2
C                                                                       SORTER     3
C     THIS ROUTINE SORTS THE COORDINATE ARRAYS AND THE STATUS ARRAY.    SORTER     4
C     THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT            SORTER     5
C     VARIABLE.                                                         SORTER     6
C                                                                       SORTER     7
C     NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.      SORTER     8
C          IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE     SORTER     9
C               THREE ARRAYS (X,Y, AND NREC) ARE SORTED SO THAT THE     SORTER    10
C               X - COORDINATE ARRAY IS IN ASCENDING ORDER.             SORTER    11
C               IF IV = 1HY, THEN THE THREE ARRAYS ARE SORTED SO        SORTER    12
C               THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER.    SORTER    13
C                                                                       SORTER    14
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      DO 10 L=1,NPOINTS,L                                               SORTER    16
      M = 2 * L - 1                                                     SORTER    17
   10 CONTINUE                                                          SORTER    18
C                                                                       SORTER    19
C     START LOOP                                                        SORTER    20
C                                                                       SORTER    21
   20 M = M / 2                                                         SORTER    22
      IF(M .EQ. 0) GO TO 70                                             SORTER    23
      K = NPOINTS - M                                                   SORTER    24
      DO 60 J=1,K                                                       SORTER    25
      L = J                                                             SORTER    26
   30 IF(L .LT. 1) GO TO 60                                             SORTER    27
C                                                                       SORTER    28
C     USE THE SORT KEY.                                                 SORTER    29
C                                                                       SORTER    30
      IF(IV .EQ. 1HY) GO TO 40                                          SORTER    31
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORTER    32
      GO TO 50                                                          SORTER    33
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORTER    34
C                                                                       SORTER    35
C     EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.                   SORTER    36
C                                                                       SORTER    37
   50 TEMP = X(L)                                                       SORTER    38
      X(L) = X(L+M)                                                     SORTER    39
      X(L+M) = TEMP                                                     SORTER    40
C                                                                       SORTER    41
C     EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.                   SORTER    42
C                                                                       SORTER    43
      TEMP = Y(L)                                                       SORTER    44
      Y(L) = Y(L+M)                                                     SORTER    45
      Y(L+M) = TEMP                                                     SORTER    46
C                                                                       SORTER    47
C     EXCHANGE POSITIONS IN THE STATUS ARRAY.                           SORTER    48
C                                                                       SORTER    49
      ITEMP = NREC(L)                                                   SORTER    50
      NREC(L) = NREC(L+M)                                               SORTER    51
      NREC(L+M) = ITEMP                                                 SORTER    52
      L=L-M                                                             8JUNE76   15
      GO TO 30                                                          SORTER    54
   60 CONTINUE                                                          SORTER    55
      GO TO 20                                                          SORTER    56
   70 RETURN                                                            SORTER    57
      END                                                               SORTER    58
```

```
      SUBROUTINE SORTCK(ICRVPTR,IV,NPOINTS)                              SORTCK     2
C                                                                        SORTCK     3
C         THIS ROUTINE MOVES THE COORDINATE ARRAYS AND THE STATUS ARRAY  SORTCK     4
C         FROM THE DATA MANAGER FILE INTO CORE AND CHECKS TO SEE IF THE  SORTCK     5
C         POINTS HAVE TO BE SORTED. IF SO, THEN THEY ARE SORTED AND      SORTCK     6
C         MOVED BACK INTO THE DATA MANAGER FILE IN THEIR NEW SORT ORDER. SORTCK     7
C                                                                        SORTCK     8
C         ICRVPTR - I - THE POINTER TO THE CURVE BEAD OF THE CURVE WHICH SORTCK     9
C                   IS TO BE EXAMINED FOR SORT ORDER.                    SORTCK    10
C              IV - I - THE INDEPENDENT VARIABLE SUBMITTED BY THE USER.  SORTCK    11
C         NPOINTS - O - THE NUMBER OF POINTS IN THE COORDINATE BEADS.    SORTCK    12
C                                                                        SORTCK    13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,   /CRVBEAD/   2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                  /CRVBEAD/   3
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/   2
C                                                                        SORTCK    16
C         MOVE THE COORDINATE ARRAYS AND THE STATUS ARRAY INTO CORE.     SORTCK    17
C                                                                        SORTCK    18
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                 SORTCK    19
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                  SORTCK    20
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                   SORTCK    21
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                   SORTCK    22
      KOMPCOD = 700000000C18 + SHIFT(NPOINTS,18)                         SORTCK    23
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                   SORTCK    24
      CALL DMGET(KOMPCOD,IXPTR,X)                                        SORTCK    25
      CALL DMGET(KOMPCOD,IYPTR,Y)                                        SORTCK    26
C                                                                        SORTCK
C         CHECK TO SEE IF THE POINTS NEED TO BE SORTED. THEY SHOULD BE   SORTCK
C         SORTED WITH THE SORT KEY EQUAL TO THE INDEPENDENT VARIABLE     SORTCK    29
C         SUBMITTED BY THE USER. IF THEY ARE ALREADY SORTED THAT WAY, NO SORTCK    30
C         SORT IS NECESSARY. IF NOT, THEY MUST BE SORTED.                SORTCK    31
C                                                                        SORTCK    32
      CALL DMGET(ICINDEP,ICRVPTR,IVAR)                                   SORTCK    33
      IF(IVAR .NE. 0) GO TO 10                                           SORTCK    34
      GO TO 20                                                           29APR     18
   10 IF(IV .EQ. IVAR) RETURN                                            SORTCK    37
C                                                                        SORTCK    38
C         THE POINTS ARE NOT SORTED AS SPECIFIED BY THE USER. SORT       SORTCK    39
C         THEM AND DUPLICATE THE SORTED ARRAYS IN THE DATA MANAGER FILE. SORTCK    40
C                                                                        SORTCK    41
   20 CALL SORTER (NPOINTS,IV)                                           29APR     19
      CALL DMSET(KOMPCOD,IRECPTR,NREC)                                   SORTCK    43
      CALL DMSET(KOMPCOD,IXPTR,X)                                        SORTCK    44
      CALL DMSET(KOMPCOD,IYPTR,Y)                                        SORTCK    45
      CALL DMSET(ICINDEP,ICRVPTR,IV)                                     SORTCK    46
      RETURN                                                             SORTCK    47
      END                                                                SORTCK    48
```

```
      1                 SUBROUTINE BUILDF3(ITYPE,A,NFP,IV,NPOINTS,IFITPTR)                BUILDF8    2
                  C                                                                       BUILDF8    3
                  C       THIS ROUTINE GETS A FIT BEAD AND FILLS IT WITH INFORMATION      BUILDF8    4
                  C       ABOUT THE FIT AND THE INTERVAL OVER WHICH IT APPLIES.           BUILDF8    5
      5           C                                                                       BUILDF8    6
                  C         ITYPE - I - THE TYPE OF FIT: 10HLEASTSQUAR FOR A LEAST        BUILDF8    7
                  C                 SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND         13SEPT    12
                  C                 6HAITKEN FOR A STRAIGHT FIT.                        13SEPT    13
                  C             A - I - THE ARRAY OF FIT PARAMETERS: THE LEAST SQUARES   BUILDF8    9
     10           C                 POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT AND BUILDF8   10
                  C                 THE ARRAY OF DERIVATIVES FOR A SPLINE FIT.          BUILDF8   11
                  C                 A DUMMY FOR THE STRAIGHT FIT.                       13SEPT    14
                  C           NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF     BUILDF8   12
                  C                 COEFFICIENTS FOR A LEAST SQUARES FIT AND THE NUMBER BUILDF8   13
     15           C                 OF POINTS FOR A SPLINE FIT.                         BUILDF8   14
                  C                 NUMBER OF DEGREE OF FIT FOR STRAIGHT.               13SEPT    15
                  C            IV - I - THE USER SUBMITTED INDEPENDENT VARIABLE.        BUILDF8   15
                  C       NPOINTS - I - THE NUMBER OF POINTS ON THE PLOT.               BUILDF8   16
                  C       IFITPTR - O - THE POINTER TO THE FIT BEAD.                    BUILDF8   17
     20           C                                                                       9BUILDF8   18
                       COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                   /AXSCALE/    2
                       COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NWEND(10),IDSPEC(19),IDCF(41),/DISPLAY/    2
                      X                IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),       6JULY77    2
                      X                IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT    1
     25               COMMON/FITBEAD/IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA               14APR76    1
                       COMMON/POINTS/X(248),Y(248),NREC(248)                            /POINTS/    2
                  C                                                                       BUILDF8   23
                  C       GET A FIT BEAD. THE LENGTH OF THE FIT                         BUILDF8   24
                  C       BEAD DEPENDS UPON THE TYPE OF FIT.                            BUILDF8   25
     30           C                                                                       BUILDF8   26
                       IF(ITYPE .EQ. 10HLEASTSQUAR) CALL DMGTBD(NFP + 5,IFITPTR)        BUILDF8   27
                       IF(ITYPE .EQ. 6HSPLINE) CALL DMGTBD(3 * NFP + 5,IFITPTR)         BUILDF8   28
                       IF (ITYPE.EQ.6HAITKEN) CALL DMGTBD (2*NPOINTS+6,IFITPTR)         20SEPT    63
                  C                                                                       BUILDF8   29
     35           C       STORE THE FIT TYPE.                                           BUILDF8   30
                  C                                                                       BUILDF8   31
                       CALL DMSET(IFTYPE,IFITPTR,ITYPE)                                 BUILDF8   32
                  C                                                                       BUILDF8   33
                  C       STORE THE INTERVAL AND INCREMENT INFORMATION.                  14APR76   62
     40           C                                                                       BUILDF8   35
                       IF(IV .EQ. 1HY) GO TO 10                                         BUILDF8   36
                       CALL DMSET(IFIVVFP,IFITPTR,X(1))                                 BUILDF8   37
                       CALL DMSET(IFIVVLP,IFITPTR,X(NPOINTS))                           BUILDF8   38
                       CALL DMSET(IFIVI,IFITPTR,XINCR/20.)                              BUILDF8   39
     45                GO TO 20                                                         BUILDF8   40
                    10 CALL DMSET(IFIVVFP,IFITPTR,Y(1))                                 BUILDF8   41
                       CALL DMSET(IFIVVLP,IFITPTR,Y(NPOINTS))                           BUILDF8   42
                       CALL DMSET(IFIVI,IFITPTR,YINCR/20.)                              BUILDF8   43
                  C                                                                       BUILDF8   44
     50           C       STORE THE NUMBER OF FIT PARAMETERS                            BUILDF8   45
                  C       AND THE ARRAY OF FIT PARAMETERS.                              BUILDF8   46
                  C                                                                       BUILDF8   47
                    20 IF (ITYPE.EQ.6HAITKEN) GO TO 30                                  20SEPT    64
                       CALL DMSET (IFNPARA,IFITPTR,NFP)                                 20SEPT    65
     55                KOMPCOD = 700000000066 + SHIFT(NFP,18)                           14APR76   63
                       CALL DMSET(KOMPCOD,IFITPTR,A)                                    BUILDF8   50
                  C                                                                       BUILDF8   51
```

```
      C          A SPLINE FIT IS NOT UNIQUELY DEFINED UNLESS THE POINTS ARRAYS  BUILDF8    52
      C          ARE KNOWN. ACCORDINGLY, IF THE FIT IS A SPLINE, STORE THE      BUILDF8    53
   60 C          X-COORDINATE ARRAY AND THE Y-COORDINATE ARRAY IN THE FIT BEAC  BUILDF8    54
      C          IN ADDITION TO THE ARRAY OF FIT PARAMETERS. OTHERWISE, STORE   BUILDF8    55
      C          ONLY THE ARRAY OF FIT PARAMETERS.                              BUILDF8    56
      C                                                                         BUILDF8    57
            IF(ITYPE .NE. 6HSPLINE) RETURN                                      BUILDF8    58
   65       C                                                                   BUILDF8    59
      C          STORE THE X-COORDINATE ARRAY.                                  BUILDF8    60
      C                                                                         BUILDF8    61
            KOMPCOD = KOMPCOD + NFP                                             BUILDF8    62
            CALL DMSET(KOMPCOD,IFITPTR,X)                                       BUILDF8    63
   70       C                                                                   BUILDF8    64
      C          STORE THE Y-COORDINATE ARRAY.                                  BUILDF8    65
      C                                                                         BUILDF8    66
            KOMPCOD = KOMPCOD + NFP                                             BUILDF8    67
            CALL DMSET(KOMPCOD,IFITPTR,Y)                                       BUILDF8    68
   75       RETURN                                                              BUILDF8    69
      C                                                                         20SEPT     66
      C          AITKEN FIT -- STORE NPOINTS, X AND Y ARRAYS                    20SEPT     67
      C                                                                         20SEPT     68
         30 CALL DMSET (IFNPARA,IFITPTR,NPOINTS)                                20SEPT     69
   80       CALL DMSET (IFNPARA+1,IFITPTR,NFP)                                  20SEPT     70
            KOMPCOD = 700000000078+SHIFT(NPOINTS,18)                            20SEPT     71
            CALL DMSET (KOMPCOD,IFITPTR,X)                                      20SEPT     72
            KOMPCOD=KOMPCOD+NPOINTS                                             20SEPT     73
            CALL DMSET (KOMPCOD,IFITPTR,Y)                                      20SEPT     74
   85       RETURN                                                              20SEPT     75
            END                                                                 BUILDF8    70
```

```
      1                 OVERLAY (6,3)                                                    INHALE     79
                        PROGRAM SPLINE                                                   INHALE     80
                  C                                                                      SPLINE      3
                  C       THIS ROUTINE SUPERVISES THE CONSTRUCTION, STORAGE, AND DISPLAY SPLINE      4
      5           C       OF SPLINE FITS. PRIMARY INPUT TO THE ROUTINE IS THE            SPLINE      5
                  C       CURVE DISPLAY ACCESS TABLE. PRIMARY OUTPUT IS FITBEADS         SPLINE      6
                  C       AND DISPLAYED CURVE FITS FOR EACH OF THE ACTIVE CURVES         SPLINE      7
                  C       LISTED IN THE CURVE DISPLAY ACCESS TABLE.                      SPLINE      8
                  C                                                                      SPLINE      9
     10                   COMMON/DISPLAY/IOSTOP,IODAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/  2
                     X                 IOAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),         6JULY77    2
                     X                 IOINSTF(19),ICDATAB(20,4,2),NCIP,IOBLOCK(20),NCIPM 1SEPT     1
                        COMMON/FITBEAD/IFTYPE,IFIVVEP,IFIVVLP,IFIVI,IFNPARA              14APR76    1
                        COMMON /SP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                 CYBTTT     1
     15                 X     ISTR,IPUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO               6JULY77    3
                        COMMON/POINTS/X(248),Y(248),NREC(248)                            /POINTS/   2
                        DIMENSION DERIV(249), W(742)                                     SPLINE     14
                        CALL ORIGIN(-5.,-3.)                                             SPLINE     15
                  C                                                                      SPLINE     16
     20           C       GET THE INDEPENDENT VARIABLE FROM THE USER.                    SPLINE     17
                  C                                                                      SPLINE     18
                        CALL GIEOM (NCON,IEOM,IBUT,253)                                  CYBTTT     10
                        CALL GETIV(IV)                                                   SPLINE     20
                        CALL GIEOM(NCON)                                                 SPLINE     21
     25           C                                                                      SPLINE     22
                  C       LOOP THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE.       SPLINE     23
                  C       IF A CURVE IS ACTIVE, SKIP IT. OTHERWISE, CURVE FIT IT.        SPLINE     24
                  C                                                                      SPLINE     25
                        DO 20 I=1,NCIP                                                   SPLINE     26
     30                 IF(ICDATAB(I,1,1) .EQ. 8HINACTIVE) GO TO 20                     SPLINE     27
                        IF(ICDATAB(I,1,2) .EQ. 8HINACTIVE) GO TO 10                      SPLINE     28
                        CALL TERASE(IDCF(ICDATAB(I,2,2)),ICDATAB(I,3,2))                 SPLINE     29
                        CALL CMRLAD(ICDATAB(I,4,2))                                      SPLINE     30
                  C                                                                      SPLINE     31
     35           C       MOVE THE POINTS ARRAYS INTO CORE AND, IF NECESSARY, SORT THEM. SPLINE     32
                  C                                                                      SPLINE     33
                     10 CALL SORTCK(ICDATAB(I,4,1),IV,NPOINTS)                           5MAY76     64
                  C                                                                      SPLINE     35
                  C       CLEAN UP THE IN-CORE POINTS ARRAYS BY REMOVING ALL POINTS      SPLINE     36
     40           C       FROM THEM EXCEPT THOSE DISPLAYED ON THE SCREEN.                SPLINE     37
                  C                                                                      SPLINE     38
                        CALL MRCLEAN(NPOINTS)                                            SPLINE     39
                        IF (NPOINTS.LE.1) CALL AERTRN                                    INHALE     61
                  C                                                                      SPLINE     40
     45           C       GET A SPLINE FIT. THE SPLINE PARAMETERS (DERIVATIVES)          SPLINE     41
                  C       ARE RETURNED IN ARRAY "DERIV".                                 SPLINE     42
                  C                                                                      SPLINE     43
                        IF(IV .EQ. 1HX) CALL SPLINC(NPOINTS,X,Y,DERIV,W,0,DUM1,DUM2)     SPLINE     44
                        IF(IV .EQ. 1HY) CALL SPLINC(NPOINTS,Y,X,DERIV,W,0,DUM1,DUM2)     SPLINE     45
     50                 C                                                               SPLINE     46
                  C      BUILD A FIT BEAD.                                              SPLINE     47
                  C                                                                      SPLINE     48
                        CALL BUILDF8(6HSPLINE,DERIV,NPOINTS,IV,NPOINTS,IFITPTR)          SPLINE     49
                  C                                                                      SPLINE     50
     55           C       DISPLAY THE FIT.                                               SPLINE     51
                  C                                                                      SPLINE     52
                        CALL IMSET(IFIVVEP,IFITPTR,TMIN)                                 SPLINE     53
```

```
      CALL DMGET(IFIVVLP,IFITPTR,TMAX)                                  SPLINE    54
      CALL DMGET(IFIVI,IFITPTR,TINCR)                                   SPLINE    55
60    CALL DCRVFIT(6HSPLINE,IV,X,Y,NPOINTS,DERIV,NPOINTS,TMIN,TMAX,     SPLINE    56
     X          TINCR,INITSS,NUMDISP,I)                                28MAY76    4
C                                                                       SPLINE    58
C         UPDATE THE CURVE DISPLAY ACCESS TABLE.                        SPLINE    59
C                                                                       SPLINE    60
65    CALL UCDATAB(I,2,6HACTIVE,INITSS,NUMDISP,IFITPTR)                 SPLINE    61
   20 CONTINUE                                                          SPLINE    62
      CALL AERTRN                                                       INHALE    82
      END                                                               SPLINE    64
```

```
      SUBROUTINE GETIV(IV)                                              GETIV      2
C                                                                       GETIV      3
C         THIS ROUTINE SOLICITS THE INDEPENDENT VARIABLE FROM THE USER, GETIV      4
C         CHECKS TO SEE THAT THE SUBMITTED INDEPENDENT VARIABLE IS      GETIV      5
C         ACCEPTABLE, AND RETURNS IT TO THE CALLING PROGRAM.            GETIV      6
C                                                                       GETIV      7
      COMMON /GF/ IBUF(128),NCON,KBYTE,MBYTE,IGN,ISIN,                  CY8777     1
     X    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                /PARMSV/   2
C                                                                       GETIV     10
C         DISPLAY THE PROMPTING MESSAGE AND LIGHT REGISTER, RETRIEVE    GETIV     11
C         THE INDEPENDENT VARIABLE, AND ERASE THE PROMPTING MESSAGE.    GETIV     12
C                                                                       GETIV     13
   10 CALL ANNOTA(1.86,-3.0,44HENTER THE INDEPENDENT VARIABLE (X OR Y) - GETIV     14
     X-- ,44,IDMESG,IGN,-0,-0,-0)                                       GETIV     15
      IX = IFIX(7.14 * DGUPIN + XORDGU + .5)                            GETIV     16
      IY = IFIX(-3.0 * DGUPIN + YORDGU + .5)                            GETIV     17
      NC=2                                                              CY8777     8
      CALL GIANS(NCON,NC,IX,IY)                                         GETIV     19
   20 CALL GIBUT(0,NCON,IDOT,IDOC)                                      GETIV     20
      IF(IDOC .EQ. 254) CALL STOP                                       14APR76   74
      IF(IDOC .NE. 253) GO TO 20                                        14APR76   75
      CALL GIANE(NCON,NC,IV)                                            GETIV     23
      CALL GIERAS(IDMESG)                                               GETIV     24
C                                                                       GETIV     25
C         CHECK TO SEE THAT THE SUBMITTED INDEPENDENT                   GETIV     26
C         VARIABLE IS ADMISSABLE.                                       GETIV     27
C                                                                       GETIV     28
      IF( (IV .EQ. 1HX) .OR. (IV .EQ. 1HY) ) RETURN                     GETIV     29
      CALL ERPORM(1.0,-3.0,29HYOU DID NOT INPUT "X" OR "Y".,29,         GETIV     30
     Y            5HGETIV,6HINFORM)                                    GETIV     31
      GO TO 10                                                          GETIV     32
      END                                                               GETIV     33
```

```
      SUBROUTINE MRCLEAN(NPOINTS)                                        MRCLEAN    2
C                                                                        MRCLEAN    3
C     THIS ROUTINE CLEANS UP THE CORE RESIDENT POINTS ARRAYS BY          MRCLEAN    4
C     REMOVING INACTIVE POINTS AND POINTS WHICH ARE OFF SCALE FROM       MRCLEAN    5
C     THE IN-CORE POINTS ARRAYS. PRIMARY INPUTS TO THE ROUTINE ARE       MRCLEAN    6
C     THE POINTS ARRAYS, THE STATUS ARRAY, AND THE SCALE INFORMATION     MRCLEAN    7
C     THE PRIMARY OUTPUT FROM THE ROUTINE IS SANITIZED POINTS ARRAYS     MRCLEAN    8
C     WHICH ARE READY TO BE FED TO THE CURVE FITTING ROUTINES.           MRCLEAN    9
C     NOTE---THE DATA MANAGER FILE CONTINUES TO HOLD THE                 MRCLEAN   10
C           UNSANITIZED ARRAYS.                                          MRCLEAN   11
C                                                                        MRCLEAN   12
C     NPOINTS - I/O - WHEN NPOINTS IS INPUT, IT IS EQUAL TO THE          MRCLEAN   13
C               NUMBER OF POINTS IN THE COORDINATE BEADS. WHEN           MRCLEAN   14
C               NPOINTS IS OUTPUT, IT IS EQUAL TO THE NUMBER OF          MRCLEAN   15
C               POINTS WHICH ARE DISPLAYED ON THE SCREEN. THE            MRCLEAN   16
C               TWO ARE NOT NECESSARILY THE SAME.                        MRCLEAN   17
C                                                                        MRCLEAN   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                     /AXSCALE/  2
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/   2
C                                                                        MRCLEAN   21
C     NPNOP --- THE NUMBER OF POINTS NOT ON THE PLOT.                    MRCLEAN   22
C                                                                        MRCLEAN   23
      NPNOP = 0                                                          MRCLEAN   24
      DO 20 I=1,NPOINTS                                                  MRCLEAN   25
      IF(NREC(I) .LT. 0) GO TO 10                                        MRCLEAN   26
      IF( (X(I) .LT. XMIN) .OR. (X(I) .GT. XMAX) ) GO TO 10              MRCLEAN   27
      IF( (Y(I) .LT. YMIN) .OR. (Y(I) .GT. YMAX) ) GO TO 10              MRCLEAN
C                                                                        MRCLEAN
C     THE "I-TH" POINT IS ON THE PLOT(CURRENTLY DISPLAYED ON THE         MRCLEAN   30
C     SCREEN). IF ANY POINTS PRIOR TO THE "I-TH" POINT ARE NOT ON        MRCLEAN   31
C     THE PLOT, THEN MOVE THE "I-TH" POINT INTO THE GAP CREATED          MRCLEAN   32
C     BY THE MISSING POINT(S). OTHERWISE, GO TO THE "(I+1)-ST" POINT     MRCLEAN   33
C                                                                        MRCLEAN   34
      IF(NPNOP .EQ. 0) GO TO 20                                          MRCLEAN   35
      X(I-NPNOP) = X(I)                                                  MRCLEAN   36
      Y(I-NPNOP) = Y(I)                                                  MRCLEAN   37
      GO TO 20                                                           MRCLEAN   38
C                                                                        MRCLEAN   39
C     THE POINT IS NOT ON THE PLOT.                                      MRCLEAN   40
C                                                                        MRCLEAN   41
   10 NPNOP = NPNOP + 1                                                  MRCLEAN   42
   20 CONTINUE                                                           MRCLEAN   43
      NPOINTS = NPOINTS - NPNOP                                          MRCLEAN   44
      RETURN                                                             MRCLEAN   45
      END                                                                MRCLEAN   46
```

```
      SUBROUTINE SORTER(NPOINTS,IV)                                     SORTER      2
C                                                                       SORTER      3
C         THIS ROUTINE SORTS THE COORDINATE ARRAYS AND THE STATUS ARRAY. SORTER      4
C         THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT        SORTER      5
C         VARIABLE.                                                     SORTER      6
C                                                                       SORTER      7
C         NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.  SORTER      8
C              IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE SORTER      9
C                   THREE ARRAYS (X,Y, AND NREC) ARE SORTED SO THAT THE SORTER     10
C                   X - COORDINATE ARRAY IS IN ASCENDING ORDER.         SORTER     11
C                   IF IV = 1HY, THEN THE THREE ARRAYS ARE SORTED SO    SORTER     12
C                   THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER. SORTER     13
C                                                                       SORTER     14
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/    2
      DO 10 L=1,NPOINTS,L                                               SORTER     16
      M = 2 * L - 1                                                     SORTER     17
   10 CONTINUE                                                          SORTER     18
C                                                                       SORTER     19
C         START LOOP                                                    SORTER     20
C                                                                       SORTER     21
   20 M = M / 2                                                         SORTER     22
      IF(M .EQ. 0) GO TO 70                                             SORTER     23
      K = NPOINTS - M                                                   SORTER     24
      DO 60 J=1,K                                                       SORTER     25
      L = J                                                             SORTER     26
   30 IF(L .LT. 1) GO TO 60                                             SORTER     27
C                                                                       SORTER     28
C       USE THE SORT KEY.                                               SORTER     29
C                                                                       SORTER     30
      IF(IV .EQ. 1HY) GO TO 40                                          SORTER     31
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORTER     32
      GO TO 50                                                          SORTER     33
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORTER     34
C                                                                       SORTER     35
C         EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.               SORTER     36
C                                                                       SORTER     37
   50 TEMP = X(L)                                                       SORTER     38
      X(L) = X(L+M)                                                     SORTER     39
      X(L+M) = TEMP                                                     SORTER     40
C                                                                       SORTER     41
C         EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.               SORTER     42
C                                                                       SORTER     43
      TEMP = Y(L)                                                       SORTER     44
      Y(L) = Y(L+M)                                                     SORTER     45
      Y(L+M) = TEMP                                                     SORTER     46
C                                                                       SORTER     47
C         EXCHANGE POSITIONS IN THE STATUS ARRAY.                       SORTER     48
C                                                                       SORTER     49
      ITEMP = NREC(L)                                                   SORTER     50
      NREC(L) = NREC(L+M)                                               SORTER     51
      NREC(L+M) = ITEMP                                                 SORTER     52
      L=L-M                                                             8JUNE76    15
      GO TO 30                                                          SORTER     54
   60 CONTINUE                                                          SORTER     55
      GO TO 20                                                          SORTER     56
   70 RETURN                                                            SORTER     57
      END                                                               SORTER     58
```

```
      SUBROUTINE SORTCK(ICRVPTR,IV,NPOINTS)                                  SORTCK     2
C                                                                            SORTCK     3
C         THIS ROUTINE MOVES THE COORDINATE ARRAYS AND THE STATUS ARRAY      SORTCK     4
C         FROM THE DATA MANAGER FILE INTO CORE AND CHECKS TO SEE IF THE      SORTCK     5
C         POINTS HAVE TO BE SORTED. IF SO, THEN THEY ARE SORTED AND          SORTCK     6
C         MOVED BACK INTO THE DATA MANAGER FILE IN THEIR NEW SORT ORDER.     SORTCK     7
C                                                                            SORTCK     8
C         ICRVPTR - I - THE POINTER TO THE CURVE BEAD OF THE CURVE WHICH     SORTCK     9
C                       IS TO BE EXAMINED FOR SORT ORDER.                    SORTCK    10
C              IV - I - THE INDEPENDENT VARIABLE SUBMITTED BY THE USER.      SORTCK    11
C         NPOINTS - O - THE NUMBER OF POINTS IN THE COORDINATE BEADS.        SORTCK    12
C                                                                            SORTCK    13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,       /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                      /CRVBEAD/  3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/   2
C                                                                            SORTCK    16
C         MOVE THE COORDINATE ARRAYS AND THE STATUS ARRAY INTO CORE.         SORTCK    17
C                                                                            SORTCK    18
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                     SORTCK    19
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                      SORTCK    20
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                       SORTCK    21
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                       SORTCK    22
      KOMPCOD = 700000000018 + SHIFT(NPOINTS,18)                             SORTCK    23
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    24
      CALL DMGET(KOMPCOD,IXPTR,X)                                            SORTCK    25
      CALL DMGET(KOMPCOD,IYPTR,Y)                                            SORTCK    26
C                                                                            SORTCK    27
C         CHECK TO SEE IF THE POINTS NEED TO BE SORTED. THEY SHOULD BE       SORTCK    28
C         SORTED WITH THE SORT KEY EQUAL TO THE INDEPENDENT VARIABLE         SORTCK    29
C         SUBMITTED BY THE USER. IF THEY ARE ALREADY SORTED THAT WAY, NO     SORTCK    30
C         SORT IS NECESSARY. IF NOT, THEY MUST BE SORTED.                    SORTCK    31
C                                                                            SORTCK    32
      CALL DMGET(ICINDEP,ICRVPTR,IVAR)                                       SORTCK    33
      IF(IVAR .NE. 0) GO TO 10                                               SORTCK    34
      GO TO 20                                                               29APR     18
   10 IF(IV .EQ. IVAR) RETURN                                                SORTCK    37
C                                                                            SORTCK    38
C         THE POINTS ARE NOT SORTED AS SPECIFIED BY THE USER. SORT           SORTCK    39
C         THEM AND DUPLICATE THE SORTED ARRAYS IN THE DATA MANAGER FILE.     SORTCK    40
C                                                                            SORTCK    41
   20 CALL SORTER (NPOINTS,IV)                                               29APR     19
      CALL DMSET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    43
      CALL DMSET(KOMPCOD,IXPTR,X)                                            SORTCK    44
      CALL DMSET(KOMPCOD,IYPTR,Y)                                            SORTCK    45
      CALL DMSET(ICINDEP,ICRVPTR,IV)                                         SORTCK    46
      RETURN                                                                 SORTCK    47
      END                                                                    SORTCK    48
```

```
      SUBROUTINE BUILDF8(ITYPE,A,NFP,IV,NPOINTS,IFITPTR)                BUILDF8    2
C                                                                       BUILDF8    3
C     THIS ROUTINE GETS A FIT BEAD AND FILLS IT WITH INFORMATION        BUILDF8    4
C     ABOUT THE FIT AND THE INTERVAL OVER WHICH IT APPLIES.             BUILDF8    5
C                                                                       BUILDF8    6
C       ITYPE - I - THE TYPE OF FIT; 10HLEASTSQUAR FOR A LEAST          BUILDF8    7
C               SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND             13SEPT    12
C               6HAITKEN FOR A STRAIGHT FIT.                            13SEPT    13
C           A - I - THE ARRAY OF FIT PARAMETERS; THE LEAST SQUARES      BUILDF8    9
C               POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT AND     BUILDF8   10
C               THE ARRAY OF DERIVATIVES FOR A SPLINE FIT.              BUILDF8   11
C               A DUMMY FOR THE STRAIGHT FIT.                           13SEPT    14
C         NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF         BUILDF8   12
C               COEFFICIENTS FOR A LEAST SQUARES FIT AND THE NUMBER     BUILDF8   13
C               OF POINTS FOR A SPLINE FIT.                             BUILDF8   14
C               NUMBER OF DEGREE OF FIT FOR STRAIGHT.                   13SEPT    15
C          IV - I - THE USER SUBMITTED INDEPENDENT VARIABLE.            BUILDF8   15
C       NPOINTS - I - THE NUMBER OF POINTS ON THE PLOT.                 BUILDF8   16
C       IFITPTR - O - THE POINTER TO THE FIT BEAD.                      BUILDF8   17
C                                                                       BUILDF8   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/  2
      COMMON/DISPLAY/IDSTOP,IODAD(80),NN,NNEND(10),IOSPEC(19),IOCF(41), /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),         6JULY77    2
     Y               IDINSTR(19),ICDA,A3(20,4,2),NCIP,IOBLOCK(20),NCIPP 1SEPT     1
      COMMON/FITBEAD/IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA               14APR76    1
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
C                                                                       BUILDF8
C     GET A FIT BEAD. THE LENGTH OF THE FIT                             BUILDF8
C     BEAD DEPENDS UPON THE TYPE OF FIT.                                BUILDF8   25
C                                                                       BUILDF8   26
      IF(ITYPE .EQ. 10HLEASTSQUAR) CALL DMGTBD(NFP + 5,IFITPTR)         BUILDF8   27
      IF(ITYPE .EQ. 6HSPLINE) CALL DMGTBD(3 * NFP + 5,IFITPTR)          BUILDF8   28
      IF (ITYPE.EQ.6HAITKEN) CALL DMGTBD (2*NPOINTS+6,IFITPTR)          20SEPT    63
C                                                                       BUILDF8   29
C     STORE THE FIT TYPE.                                               BUILDF8   30
C                                                                       BUILDF8   31
      CALL DMSET(IFTYPE,IFITPTR,ITYPE)                                  BUILDF8   32
C                                                                       BUILDF8   33
C     STORE THE INTERVAL AND INCREMENT INFORMATION.                     14APR76   62
C                                                                       BUILDF8   35
      IF(IV .EQ. 1HY) GO TO 10                                          BUILDF8   36
      CALL DMSET(IFIVVFP,IFITPTR,X(1))                                  BUILDF8   37
      CALL DMSET(IFIVVLP,IFITPTR,X(NPOINTS))                            BUILDF8   38
      CALL DMSET(IFIVI,IFITPTR,XINCR/20.)                               BUILDF8   39
      GO TO 20                                                          BUILDF8   40
   10 CALL DMSET(IFIVVFP,IFITPTR,Y(1))                                  BUILDF8   41
      CALL DMSET(IFIVVLP,IFITPTR,Y(NPOINTS))                            BUILDF8   42
      CALL DMSET(IFIVI,IFITPTR,YINCR/20.)                               BUILDF8   43
C                                                                       BUILDF8   44
C     STORE THE NUMBER OF FIT PARAMETERS                                BUILDF8   45
C     AND THE ARRAY OF FIT PARAMETERS.                                  BUILDF8   46
C                                                                       BUILDF8   47
   20 IF (ITYPE.EQ.6HAITKEN) GO TO 30                                   20SEPT    64
      CALL DMSET (IFNPARA,IFITPTR,NFP)                                  20SEPT    65
      KOMCOD = 700000000000B + SHIFT(NFP,18)                            14APR76   63
      CALL DMSET(KOMCOD,IFITPTR,A)                                      BUILDF8   50
C                                                                       BUILDF8   51
```

```
      C          A SPLINE FIT IS NOT UNIQUELY DEFINED UNLESS THE POINTS ARRAYS  BUILDF8    52
      C          ARE KNOWN. ACCORDINGLY, IF THE FIT IS A SPLINE, STORE THE      BUILDF8    53
   60 C          X-COORDINATE ARRAY AND THE Y-COORDINATE ARRAY IN THE FIT BEAC  BUILDF8    54
      C          IN ADDITION TO THE ARRAY OF FIT PARAMETERS. OTHERWISE, STORE   BUILDF8    55
      C          ONLY THE ARRAY OF FIT PARAMETERS.                              BUILDF8    56
      C                                                                         BUILDF8    57
            IF(ITYPE .NE. 6HSPLINE) RETURN                                      BUILDF8    58
   65 C                                                                         BUILDF8    59
      C          STORE THE X-COORDINATE ARRAY.                                  BUILDF8    60
      C                                                                         BUILDF8    61
            KOMPCOD = KOMPCOD + NFP                                             BUILDF8    62
            CALL DMSET(KOMPCOD,IFITPTR,X)                                       BUILDF8    63
   70 C                                                                         BUILDF8    64
      C          STORE THE Y-COORDINATE ARRAY.                                  BUILDF8    65
      C                                                                         BUILDF8    66
            KOMPCOD = KOMPCOD + NFP                                             BUILDF8    67
            CALL DMSET(KOMPCOD,IFITPTR,Y)                                       BUILDF8    68
   75       RETURN                                                             BUILDF8    69
      C                                                                         20SEPT     66
      C          AITKEN FIT -- STORE NPOINTS, X AND Y ARRAYS                    20SEPT     67
      C                                                                         20SEPT     68
         30 CALL DMSET (IFNPARA,IFITPTR,NPOINTS)                                20SEPT     69
   80    CALL DMSET (IFNPARA+1,IFITPTR,NFP)                                     20SEPT     70
         KOMPCOD = 700000000078+SHIFT(NPOINTS,18)                               20SEPT     71
         CALL DMSET (KOMPCOD,IFITPTR,X)                                         20SEPT     72
         KOMPCOD=KOMPCOD+NPOINTS                                                20SEPT     73
         CALL DMSET (KOMPCOD,IFITPTR,Y)                                         20SEPT     74
   85    RETURN                                                                 20SEPT     75
         END                                                                    BUILDF8    70
```

```
      SUBROUTINE SPLINC(N,XN,FN,GN,W,ISW,RR1,RRN)                        SPLINC     2
C                                                                        SPLINC     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLINC     4
C                                                                        SPLINC     5
C  CUBIC SPLINE INTERPOLATION ROUTINE.                                   SPLINC     6
C                                                                        SPLINC     7
C    N   = NUMBER OF DATA POINTS (I.E. THE NUMBER OF KNOTS)              SPLINC     8
C    XN  = ARRAY OF X-VALUES                                             SPLINC     9
C    FN  = ARRAY OF FUNCTION VALUES. (I.E. REQUIRED SPLINE VALUES)       SPLINC    10
C    GN  = ARRAY CONTAINING THE VALUES OF THE FIRST DERIVATIVE OF THE    SPLINC    11
C          SPLINE AT EACH KNOT. (RETURNED BY SPLINC)                     SPLINC    12
C    W   = A WORKING ARRAY DIMENSIONED 3*N-2                             SPLINC    13
C                                                                        SPLINC    14
C    ISW = 0 NO BOUNDARY CONDITIONS SPECIFIED                            SPLINC    15
C        = 1 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(1)          SPLINC    16
C        = 2 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(N)          SPLINC    17
C        = 3 USER SUPPLIES BOTH SECOND DERIVATIVE VALUES                 SPLINC    18
C        = 4 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(1)           SPLINC    19
C        = 5 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(N)           SPLINC    20
C        = 6 USER SUPPLIES BOTH FIRST DERIVATIVE VALUES                  SPLINC    21
C                                                                        SPLINC    22
C    RR1 = BOUNDARY VALUE AT XN(1)                                       SPLINC    23
C    RRN = BOUNDARY VALUE AT XN(N)                                       SPLINC    24
C                                                                        SPLINC    25
C                                                                        SPLINC    26
C          CONDITIONS:                                                   SPLINC    27
C            1  N .GE. 3                                                 SPLINC    28
C            2  MUST HAVE XN(1) .LT. XN(2) .LT. ... .LT. XN(N)           SPLINC    29
C                                                                        SPLINC    30
C                                                                        SPLINC    31
C          THE RESULTS OF SPLINC CAN BE COUPLED WITH SPLIN2 TO EVALUATE  SPLINC    32
C          THE SPLINE OR ITS FIRST DERIVATIVE FOR ANY X WHICH SATISFIES  SPLINC    33
C            XN(1) .GE. X .LE. XN(N)                                     SPLINC    34
C                                                                        SPLINC    35
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLINC    36
C                                                                        SPLINC    37
C                                                                        SPLINC    38
      DIMENSION W(1),XN(1) ,FN(1),GN(1)                                  SPLINC    39
      IK=ISW+1                                                           SPLINC    40
      GO TO(1,2,3,4,60,70,80),IK                                         SPLINC    41
    1 R1=0.                                                              SPLINC    42
      RN=0.                                                              SPLINC    43
      GO TO 11                                                           SPLINC    44
    2 R1=RR1                                                             SPLINC    45
      RN=0.                                                              SPLINC    46
      GO TO 11                                                           SPLINC    47
    3 R1=0.                                                              SPLINC    48
      RN=RRN                                                             SPLINC    49
      GO TO 11                                                           SPLINC    50
    4 R1=RR1                                                             SPLINC    51
      RN=RRN                                                             SPLINC    52
   11 W(2)=1.0/(XN(2)-XN(1))                                             SPLINC    53
      W(1)=0.*W(2)                                                       SPLINC    54
      GN(1)=3.*(FN(2)-FN(1))*W(2)**2-R1/2.                               SPLINC    55
      K=3                                                                SPLINC    56
      J=2                                                                SPLINC    57
      W(K)=1.0/(XN(J)-XN(J-1))                                           SPLINC    58
```

```
      5 W(K+2)=1.0/(XN(J+1)-XN(J))                                          SPLINC     59
        W(K+1)=2.*(W(K)+W(K+2))                                             SPLINC     60
   60   GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2       SPLINC     61
        K=K+3                                                               SPLINC     62
        J=J+1                                                               SPLINC     63
        IF(J .EQ. N  ) GO TO 8                                              SPLINC     64
        W(K)=W(K-1)                                                         SPLINC     65
   65   GO TO 5                                                             SPLINC     66
      8 W(K)=1.0/(XN(N)-XN(N-1))                                            SPLINC     67
        W(K+1)=2.*W(K)                                                      SPLINC     68
        GN(J)=3.*(FN(N)-FN(N-1))*W(K)**2+RN/2.                              SPLINC     69
C                                                                           SPLINC     70
   70   MA=3*N-2                                                            SPLINC     71
C                                                                           SPLINC     72
        DO 10 J=4,MA,3                                                      SPLINC     73
     10 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                                    SPLINC     74
        DO 20 J=3,MA,3                                                      SPLINC     75
   75   20 W(J)=W(J)/W(J-2)                                                 SPLINC     76
        IC=3                                                                SPLINC     77
        DO 30 J=2,N                                                         SPLINC     78
        GN(J)=GN(J)-W(IC)*GN(J-1)                                           SPLINC     79
     30 IC=IC+3                                                             SPLINC     80
   80   N1=N+1                                                              SPLINC     81
        IC=MA-2                                                             SPLINC     82
        GN(N)=GN(N)/W(MA)                                                   SPLINC     83
        DO 40 J=2,N                                                         SPLINC     84
        K=N1-J                                                              SPLINC
   85   GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                                 SPLINC
     40 IC=IC-3                                                             SPLINC     87
C                                                                           SPLINC     88
        RETURN                                                              SPLINC     89
     60 H=XN(2)-XN(1)                                                       SPLINC     90
   90   W(2)=1./(XN(3)-XN(2))                                               SPLINC     91
        W(1)=2.*(W(2)+1.0/H)                                                SPLINC     92
        GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H          SPLINC     93
        K=3                                                                 SPLINC     94
        J=2                                                                 SPLINC     95
   95   W(K)=W(K-1)                                                         SPLINC     96
     61 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                        SPLINC     97
        W(K+1)=2.0*(W(K)+W(K+2))                                            SPLINC     98
        GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2     SPLINC     99
        J=J+1                                                               SPLINC    100
  100   K=K+3                                                               SPLINC    101
        IF(J .EQ. N-1) GO TO 62                                             SPLINC    102
        W(K)=W(K-1)                                                         SPLINC    103
        GO TO 61                                                            SPLINC    104
     62 W(K)=XN(N)-XN(N-1)                                                  SPLINC    105
  105   W(K+1)=2.*W(K)                                                      SPLINC    106
        GN(J)=3.*(FN(N)-FN(N-1))                                            SPLINC    107
        NN=N-1                                                              SPLINC    108
        IC=1                                                                SPLINC    109
        GO TO 200                                                           SPLINC    110
  110 C                                                                     SPLINC    111
     70 W(2)=1.0/(XN(2)-XN(1))                                              SPLINC    112
        W(1)=2.*W(2)                                                        SPLINC    113
        GN(1)=3.*(FN(2)-FN(1))*W(2)**2                                      SPLINC    114
        K=3                                                                 SPLINC    115
```

```
                      J=2                                                                SPLINC   116
                      W(K)=1.0/(XN(J)-XN(J-1))                                           SPLINC   117
                   71 W(K+2)=1.0/(XN(J+1)-XN(J))                                         SPLINC   118
                      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   119
                      GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2      SPLINC   120
                      K=K+3                                                              SPLINC   121
                      J=J+1                                                              SPLINC   122
                      IF(J .EQ. N-1) GO TO 72                                            SPLINC   123
                      W(K)=W(K-1)                                                        SPLINC   124
                      GO TO 71                                                           SPLINC   125
                   72 W(K)=1.0/(XN(N-1)-XN(N-2))                                         SPLINC   126
                      W(K+2)=1.0/(XN(N)-XN(N-1))                                         SPLINC   127
                      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   128
                      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2    SPLINC   129
                     1      -RRN*W(K+2)                                                 SPLINC   130
                      NN=N-1                                                             SPLINC   131
                      IK=2                                                               SPLINC   132
                      GO TO 200                                                          SPLINC   133
                C                                                                        SPLINC   134
                   80 H=XN(2)-XN(1)                                                      SPLINC   135
                      W(2)=1.0/(XN(3)-XN(2))                                             SPLINC   136
                      W(1)=2.0*W(2)+1.0/H                                                SPLINC   137
                      GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H         SPLINC   138
                      K=3                                                                SPLINC   139
                      J=2                                                                SPLINC   140
                      W(K)=W(K-1)                                                        SPLINC   141
                   81 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                       SPLINC   142
                      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   143
                      GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2    SPLINC   144
                      J=J+1                                                              SPLINC   145
                      K=K+3                                                              SPLINC   146
                      IF(J .EQ. N-2) GO TO 82                                            SPLINC   147
                      W(K)=W(K-1)                                                        SPLINC   148
                      GO TO 81                                                           SPLINC   149
                   82 W(K)=1.0/(XN(N-1)-XN(N-2))                                         SPLINC   150
                      W(K+2)=1.0/(XN(N)-XN(N-1))                                         SPLINC   151
                      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   152
                      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2    SPLINC   153
                     1      -RRN*W(K+2)                                                 SPLINC   154
                      NN=N-2                                                             SPLINC   155
                      IK=3                                                               SPLINC   156
                C                                                                        SPLINC   157
                  200 MA=3*NN-2                                                          SPLINC   158
                      DO 210 J=4,MA,3                                                    SPLINC   159
                  210 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                                   SPLINC   160
                      DO 220 J=3,MA,3                                                    SPLINC   161
                  220 W(J)=W(J)/W(J-2)                                                   SPLINC   162
                      IC=3                                                               SPLINC   163
                      DO 230 J=2,NN                                                      SPLINC   164
                      GN(J)=GN(J)-W(IC)*GN(J-1)                                          SPLINC   165
                  230 IC=IC+3                                                            SPLINC   166
                      N1=NN+1                                                            SPLINC   167
                      IC=MA-2                                                            SPLINC   168
                      GN(NN)=GN(NN)/W(MA)                                                SPLINC   169
                      DO 240 J=2,NN                                                      SPLINC   170
                      K=N1-J                                                             SPLINC   171
                      GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                                SPLINC   172
```

400

```
      240 IC=IC-3
C
          GO TO (300,320,340),IK
C
      300 N1=N+1
          DO 305 J=1,NN
      305 GN(N1-J)=GN(N1-J-1)
          GN(1)=RR1
          RETURN
      320 GN(N)=RRN
          RETURN
      340 DO 345 J=1,NN
      345 GN(N-J)=GN(N-J-1)
          GN(1)=RR1
          GN(N)=RRN
          RETURN
          END
```

```
SPLINC  173
SPLINC  174
SPLINC  175
SPLINC  176
SPLINC  177
SPLINC  178
SPLINC  179
SPLINC  180
SPLINC  181
SPLINC  182
SPLINC  183
SPLINC  184
SPLINC  185
SPLINC  186
SPLINC  187
SPLINC  188
SPLINC  189
```

```
      OVERLAY (6,4)                                                    FIXINH   26
      PROGRAM STRAIT                                                   INHALE   84
C                                                                      STRAIT    3
C        THIS ROUTINE SUPERVISES THE CONSTRUCTION, STORAGE, AND        STRAIT    4
C        DISPLAY OF THE STRAIGHT LINE FIT.  LINES ARE FIT USING INTERP. STRAIT    5
C                                                                      STRAIT    6
      COMMON/POINTS/X(248),Y(248),NREC(248)                            /POINTS/  2
      COMMON/FITREAD/IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA              14APR76   1
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41), /DISPLAY/ 2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),         6JULY77   2
     X               IDINSTR(19),ICDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT    1
      COMMON /GP/ I2UF(126),NCON,NBYTE,MBYTE,IGN,ISIN,                 CYB777    1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                6JULY77   3
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC, /CRVBEAD/ 2
     Y               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                /CRVBEAD/ 3
C                                                                      STRAIT   12
C        GET THE DEGREE AND INDEPENDENT VARIABLE FROM THE USER         STRAIT   13
C                                                                      STRAIT   14
C     NDEG=FLTIN(38HENTER THE DEGREE OF THE INTERPOLATION ,38,         20SEPT   80
C    1     -628,-1200)                                                20SEPT   81
      NDEG=1                                                           20SEPT   82
      CALL GIEOM (NCON,IEOM,IBUT,253)                                  CYB777   11
      CALL GETIV (IV)                                                  STRAIT   17
      CALL GIEOM (NCON)                                                STRAIT   18
C                                                                      STRAIT   19
C        LOOP THROUGH THE ROWS OF THE CURVE DISPLAY ACCESS TABLE.      STRAIT   20
C        IF A CURVE IS INACTIVE, SKIP IT. OTHERWISE, CURVE FIT IT      STRAIT
C                                                                      STRAIT
      DO 20 I=1,NCIP                                                   STRAIT   23
      IF (ICDATAB(I,1,1).EQ.6HINACTIVE) GO TO 20                       STRAIT   24
      IF (ICDATAB(I,1,2).EQ.6HINACTIVE) GO TO 10                       20SEPT   84
      CALL TERASE (IDCF(ICDATAB(I,2,2)),ICDATAB(I,3,2))                STRAIT   26
      CALL DMRLBD (ICDATAB(I,4,2))                                     STRAIT   27
C                                                                      STRAIT   28
C        MOVE POINTS ARRAYS INTO CORE AND SORT IF NECESSARY            STRAIT   29
C                                                                      STRAIT   30
   10 CALL SORTCK (ICDATAB(I,4,1),IV,NPOINTS)                          STRAIT   31
      CALL MRCLEAN (NPOINTS)                                           STRAIT   32
      IF (NPOINTS.LE.1) CALL AERTRN                                    INHALE   85
C                                                                      STRAIT   33
C        BUILD A FIT BEAD                                              STRAIT   34
C                                                                      STRAIT   35
      CALL BUILBFB (6HAITKEN,DUMMY,NDEG,IV,NPOINTS,IFITPTR)            20SEPT   85
      CALL DMSET (ICFIT,ICDATAB(I,4,1),IFITPTR)                        STRAIT   37
C                                                                      STRAIT   38
C        DISPLAY THE FIT                                               STRAIT   39
C                                                                      STRAIT   40
      CALL DMGET (IFIVVFP,IFITPTR,TMIN)                                STRAIT   41
      CALL DMGET (IFIVVLP,IFITPTR,TMAX)                                STRAIT   42
      CALL DMGET (IFIVI,IFITPTR,TINCR)                                 STRAIT   43
      CALL CCRVFIT (6HAITKEN,IV,X,Y,NPOINTS,DUMMY,NDEG,TMIN,TMAX,      STRAIT   44
     1     TINCR,INITSS,NUMDISP,I)                                     STRAIT   45
      CALL UCDATAB (I,2,6HACTIVE,INITSS,NUMDISP,IFITPTR)               STRAIT   46
   20 CONTINUE                                                         STRAIT   47
      CALL AERTRN                                                      INHALE   86
      END                                                              STRAIT   49
```

```
      SUBROUTINE GETIV(IV)                                              GETIV    2
C                                                                       GETIV    3
C         THIS ROUTINE SOLICITS THE INDEPENDENT VARIABLE FROM THE USER, GETIV    4
C         CHECKS TO SEE THAT THE SUBMITTED INDEPENDENT VARIABLE IS      GETIV    5
C         ACCEPTABLE, AND RETURNS IT TO THE CALLING PROGRAM.            GETIV    6
C                                                                       GETIV    7
      COMMON /GP/ IBUF(129),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYB777   1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CRTO                 6JULY77  3
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                /PARMSV/ 2
C                                                                       GETIV   10
C         DISPLAY THE PROMPTING MESSAGE AND LIGHT REGISTER, RETRIEVE    GETIV   11
C         THE INDEPENDENT VARIABLE, AND ERASE THE PROMPTING MESSAGE.    GETIV   12
C                                                                       GETIV   13
   10 CALL ANNOTA(1.86,-3.0,44HENTER THE INDEPENDENT VARIABLE (X OR Y) - GETIV   14
     X-- ,44,IDMESG,IGN,-0,-0,-0)                                       GETIV   15
      IX = IFIX(7.14 * DGUPIN + XORDGU + .5)                            GETIV   16
      IY = IFIX(-3.0 * DGUPIN + YORDGU + .5)                            GETIV   17
      NC=2                                                              CYB777   8
      CALL GIANS(NCON,NC,IX,IY)                                         GETIV   19
   20 CALL GIBUF(0,NCON,IODT,IODC)                                      GETIV   20
      IF(IODC .EQ. 254) CALL STOP                                       14APR76  74
      IF(IODC .NE. 253) GO TO 20                                        14APR76  75
      CALL GIANE(NCON,NC,IV)                                            GETIV   23
      CALL GIERAS(IDMESG)                                               GETIV   24
C                                                                       GETIV   25
C         CHECK TO SEE THAT THE SUBMITTED INDEPENDENT                   GETIV   26
C         VARIABLE IS ADMISSABLE.                                       GETIV   27
C                                                                       GETIV   28
      IF( (IV .EQ. 1HX) .OR. (IV .EQ. 1HY) ) RETURN                     GETIV   29
      CALL ERRORM(1.0,-3.0,29HYOU DID NOT INPUT "X" OR "Y".,29,         GETIV   30
     X            5HGETIV,6HINFORM)                                     GETIV   31
      GO TO 10                                                          GETIV   32
      END                                                               GETIV   33
```

```
      SUBROUTINE MRCLEAN(NPOINTS)                                          MRCLEAN     2
C                                                                          MRCLEAN     3
C         THIS ROUTINE CLEANS UP THE CORE RESIDENT POINTS ARRAYS BY        MRCLEAN     4
C         REMOVING INACTIVE POINTS AND POINTS WHICH ARE OFF SCALE FROM     MRCLEAN     5
C         THE IN-CORE POINTS ARRAYS. PRIMARY INPUTS TO THE ROUTINE ARE     MRCLEAN     6
C         THE POINTS ARRAYS, THE STATUS ARRAY, AND THE SCALE INFORMATION   MRCLEAN     7
C         THE PRIMARY OUTPUT FROM THE ROUTINE IS SANITIZED POINTS ARRAYS   MRCLEAN     8
C         WHICH ARE READY TO BE FED TO THE CURVE FITTING ROUTINES.         MRCLEAN     9
C         NOTE---THE DATA MANAGER FILE CONTINUES TO HOLD THE               MRCLEAN    10
C               UNSANITIZED ARRAYS.                                        MRCLEAN    11
C                                                                          MRCLEAN    12
C         NPOINTS - I/O - WHEN NPOINTS IS INPUT, IT IS EQUAL TO THE        MRCLEAN    13
C                   NUMBER OF POINTS IN THE COORDINATE BEADS. WHEN         MRCLEAN    14
C                   NPOINTS IS OUTPUT, IT IS EQUAL TO THE NUMBER OF        MRCLEAN    15
C                   POINTS WHICH ARE DISPLAYED ON THE SCREEN. THE          MRCLEAN    16
C                   TWO ARE NOT NECESSARILY THE SAME.                      MRCLEAN    17
C                                                                          MRCLEAN    18
      COMMON/AXSCALE/XMIN,YINCR,XMAX,YMIN,YINCR,YMAX                       /AXSCALE/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                /POINTS/    2
C                                                                          MRCLEAN    21
C         NPNOP --- THE NUMBER OF POINTS NOT ON THE PLOT.                  MRCLEAN    22
C                                                                          MRCLEAN    23
      NPNOP = 0                                                            MRCLEAN    24
      DO 20 I=1,NPOINTS                                                    MRCLEAN    25
      IF(NREC(I) .LT. 0) GO TO 10                                          MRCLEAN    26
      IF( (X(I) .LT. XMIN) .OR. (X(I) .GT. XMAX) ) GO TO 10                MRCLEAN    27
      IF( (Y(I) .LT. YMIN) .OR. (Y(I) .GT. YMAX) ) GO TO 10                MRCLEAN    28
C                                                                          MRCLEAN    29
C         THE "I-TH" POINT IS ON THE PLOT(CURRENTLY DISPLAYED ON THE       MRCLEAN    30
C         SCREEN). IF ANY POINTS PRIOR TO THE "I-TH" POINT ARE NOT ON      MRCLEAN    31
C         THE PLOT, THEN MOVE THE "I-TH" POINT INTO THE GAP CREATED        MRCLEAN    32
C         BY THE MISSING POINT(S). OTHERWISE, GO TO THE "(I+1)-ST" POINT   MRCLEAN    33
C                                                                          MRCLEAN    34
      IF(NPNOP .EQ. 0) GO TO 20                                            MRCLEAN    35
      X(I-NPNOP) = X(I)                                                    MRCLEAN    36
      Y(I-NPNOP) = Y(I)                                                    MRCLEAN    37
      GO TO 20                                                             MRCLEAN    38
C                                                                          MRCLEAN    39
C         THE POINT IS NOT ON THE PLOT.                                    MRCLEAN    40
C                                                                          MRCLEAN    41
   10 NPNOP = NPNOP + 1                                                    MRCLEAN    42
   20 CONTINUE                                                             MRCLEAN    43
      NPOINTS = NPOINTS - NPNOP                                            MRCLEAN    44
      RETURN                                                               MRCLEAN    45
      END                                                                  MRCLEAN    46
```

```
      SUBROUTINE SORTER(NPOINTS,IV)                                     SORTER     2
C                                                                       SORTER     3
C     THIS ROUTINE SORTS THE COORDINATE ARRAYS AND THE STATUS ARRAY.    SORTER     4
C     THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT            SORTER     5
C     VARIABLE.                                                         SORTER     6
C                                                                       SORTER     7
C     NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.      SORTER     8
C          IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE     SORTER     9
C               THREE ARRAYS (X,Y, AND NREC) ARE SORTED SO THAT THE     SORTER    10
C               X - COORDINATE ARRAY IS IN ASCENDING ORDER.             SORTER    11
C               IF IV = 1HY, THEN THE THREE ARRAYS ARE SORTED SO        SORTER    12
C               THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER.    SORTER    13
C                                                                       SORTER    14
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      DO 10 L=1,NPOINTS,L                                               SORTER    16
      M = 2 * L - 1                                                     SORTER    17
   10 CONTINUE                                                          SORTER    18
C                                                                       SORTER    19
C     START LOOP                                                        SORTER    20
C                                                                       SORTER    21
   20 M = M / 2                                                         SORTER    22
      IF(M .EQ. 0) GO TO 70                                             SORTER    23
      K = NPOINTS - M                                                   SORTER    24
      DO 60 J=1,K                                                       SORTER    25
      L = J                                                             SORTER    26
   30 IF(L .LT. 1) GO TO 60                                             SORTER    27
C                                                                       SORTER    28
C     USE THE SORT KEY.                                                 SORTER    29
C                                                                       SORTER    30
      IF(IV .EQ. 1HY) GO TO 40                                          SORTER    31
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORTER    32
      GO TO 50                                                          SORTER    33
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORTER    34
C                                                                       SORTER    35
C     EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.                   SORTER    36
C                                                                       SORTER    37
   50 TEMP = X(L)                                                       SORTER    38
      X(L) = X(L+M)                                                     SORTER    39
      X(L+M) = TEMP                                                     SORTER    40
C                                                                       SORTER    41
C     EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.                   SORTER    42
C                                                                       SORTER    43
      TEMP = Y(L)                                                       SORTER    44
      Y(L) = Y(L+M)                                                     SORTER    45
      Y(L+M) = TEMP                                                     SORTER    46
C                                                                       SORTER    47
C     EXCHANGE POSITIONS IN THE STATUS ARRAY.                           SORTER    48
C                                                                       SORTER    49
      ITEMP = NREC(L)                                                   SORTER    50
      NREC(L) = NREC(L+M)                                               SORTER    51
      NREC(L+M) = ITEMP                                                 SORTER    52
      L=L-M                                                             8JUNE76   15
      GO TO 30                                                          SORTER    54
   60 CONTINUE                                                          SORTER    55
      GO TO 20                                                          SORTER    56
   70 RETURN                                                            SORTER    57
      END                                                               SORTER    58
```

```
      SUBROUTINE SORTCK(ICRVPTR,IV,NPOINTS)                                  SORTCK     2
C                                                                            SORTCK     3
C     THIS ROUTINE MOVES THE COORDINATE ARRAYS AND THE STATUS ARRAY          SORTCK     4
C     FROM THE DATA MANAGER FILE INTO CORE AND CHECKS TO SEE IF THE          SORTCK     5
C     POINTS HAVE TO BE SORTED. IF SO, THEN THEY ARE SORTED AND              SORTCK     6
C     MOVED BACK INTO THE DATA MANAGER FILE IN THEIR NEW SORT ORDER.         SORTCK     7
C                                                                            SORTCK     8
C     ICRVPTR - I - THE POINTER TO THE CURVE BEAD OF THE CURVE WHICH         SORTCK     9
C               IS TO BE EXAMINED FOR SORT ORDER.                            SORTCK    10
C          IV - I - THE INDEPENDENT VARIABLE SUBMITTED BY THE USER.          SORTCK    11
C     NPOINTS - O - THE NUMBER OF POINTS IN THE COORDINATE BEADS.            SORTCK    12
C                                                                            SORTCK    13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPASAN,ICPARAV,ICNPTS,ICSYM,ICREC,       /CRVBEAD/   2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                      /CRVBEAD/   3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/    2
C                                                                            SORTCK    16
C     MOVE THE COORDINATE ARRAYS AND THE STATUS ARRAY INTO CORE.             SORTCK    17
C                                                                            SORTCK    18
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                     SORTCK    19
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                      SORTCK    20
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                       SORTCK    21
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                       SORTCK    22
      KOMPCOD = 700000000018 + SHIFT(NPOINTS,18)                             SORTCK    23
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    24
      CALL DMGET(KOMPCOD,IXPTR,X)                                            SORTCK    25
      CALL DMGET(KOMPCOD,IYPTR,Y)                                            SORTCK    26
C                                                                            SORTCK    27
C     CHECK TO SEE IF THE POINTS NEED TO BE SORTED. THEY SHOULD BE           SORTCK    28
C     SORTED WITH THE SORT KEY EQUAL TO THE INDEPENDENT VARIABLE             SORTCK    29
C     SUBMITTED BY THE USER. IF THEY ARE ALREADY SORTED THAT WAY, NO         SORTCK    30
C     SORT IS NECESSARY. IF NOT, THEY MUST BE SORTED.                        SORTCK    31
C                                                                            SORTCK    32
      CALL DMGET(ICINDEP,ICRVPTR,IVAR)                                       SORTCK    33
      IF(IVAR .NE. 0) GO TO 10                                               SORTCK    34
      GO TO 20                                                               29APR     18
   10 IF(IV .EQ. IVAR) RETURN                                                SORTCK    37
C                                                                            SORTCK    38
C     THE POINTS ARE NOT SORTED AS SPECIFIED BY THE USER. SORT               SORTCK    39
C     THEM AND DUPLICATE THE SORTED ARRAYS IN THE DATA MANAGER FILE.         SORTCK    40
C                                                                            SORTCK    41
   20 CALL SORTER (NPOINTS,IV)                                               29APR     19
      CALL DMSET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    43
      CALL DMSET(KOMPCOD,IXPTR,X)                                            SORTCK    44
      CALL DMSET(KOMPCOD,IYPTR,Y)                                            SORTCK    45
      CALL DMSET(ICINDEP,ICRVPTR,IV)                                         SORTCK    46
      RETURN                                                                 SORTCK    47
      END                                                                    SORTCK    48
```

```
      SUBROUTINE BUILDFB(ITYPE,A,NFP,IV,NPOINTS,IFITPTR)                BUILDFB    2
C                                                                       BUILDFB    3
C     THIS ROUTINE GETS A FIT BEAD AND FILLS IT WITH INFORMATION        BUILDFB    4
C     ABOUT THE FIT AND THE INTERVAL OVER WHICH IT APPLIES.             BUILDFB    5
C                                                                       BUILDFB    6
C       ITYPE - I - THE TYPE OF FIT; 10HLEASTSQUAR FOR A LEAST          BUILDFB    7
C               SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND             13SEPT    12
C               6HAITKEN FOR A STRAIGHT FIT.                            13SEPT    13
C           A - I - THE ARRAY OF FIT PARAMETERS; THE LEAST SQUARES      BUILDFB    9
C               POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT AND     BUILDFB   10
C               THE ARRAY OF DERIVATIVES FOR A SPLINE FIT.              BUILDFB   11
C               A DUMMY FOR THE STRAIGHT FIT.                           13SEPT    14
C         NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF         BUILDFB   12
C               COEFFICIENTS FOR A LEAST SQUARES FIT AND THE NUMBER     BUILDFB   13
C               OF POINTS FOR A SPLINE FIT.                             BUILDFB   14
C               NUMBER OF DEGREES OF FIT FOR STRAIGHT.                  13SEPT    15
C          IV - T - THE USER SUBMITTED INDEPENDENT VARIABLE.            BUILDFB   15
C       NPOINTS - I - THE NUMBER OF POINTS ON THE PLOT.                 BUILDFB   16
C       IFITPTR - O - THE POINTER TO THE FIT BEAD.                      BUILDFB   17
C                                                                       BUILDFB   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/  2
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),          6JULY77   2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
      COMMON/FITBEAD/IFITYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA              14APR76   1
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/  2
C                                                                       BUILDFB   23
C     GET A FIT BEAD. THE LENGTH OF THE FIT                             BUILDFB   24
C     BEAD DEPENDS UPON THE TYPE OF FIT.                                BUILDFB   25
C                                                                       BUILDFB   26
      IF(ITYPE .EQ. 10HLEASTSQUAR) CALL GMGTBD(NFP + 5,IFITPTR)         BUILDFB   27
      IF(ITYPE .EQ. 6HSPLINE) CALL GMGTBD(3 * NFP + 5,IFITPTR)          BUILDFB   28
      IF (ITYPE.EQ.6HAITKEN) CALL GMGTBD (2*NPOINTS+6,IFITPTR)          26SEPT    63
C                                                                       BUILDFB   29
C     STORE THE FIT TYPE.                                               BUILDFB   30
C                                                                       BUILDFB   31
      CALL GMSET(IFTYPE,IFITPTR,ITYPE)                                  BUILDFB   32
C                                                                       BUILDFB   33
C     STORE THE INTERVAL AND INCREMENT INFORMATION.                     14APR76   62
C                                                                       BUILDFB   35
      IF(IV .EQ. 1HY) GO TO 10                                          BUILDFB   36
      CALL GMSET(IFIVVFP,IFITPTR,X(1))                                  BUILDFB   37
      CALL GMSET(IFIVVLP,IFITPTR,X(NPOINTS))                            BUILDFB   38
      CALL GMSET(IFIVI,IFITPTR,XINCR/20.)                               BUILDFB   39
      GO TO 20                                                          BUILDFB   40
   10 CALL GMSET(IFIVVFP,IFITPTR,Y(1))                                  BUILDFB   41
      CALL GMSET(IFIVVLP,IFITPTR,Y(NPOINTS))                            BUILDFB   42
      CALL GMSET(IFIVI,IFITPTR,YINCR/20.)                               BUILDFB   43
C                                                                       BUILDFB   44
C     STORE THE NUMBER OF FIT PARAMETERS                                BUILDFB   45
C     AND THE ARRAY OF FIT PARAMETERS.                                  BUILDFB   46
C                                                                       BUILDFB   47
   20 IF (ITYPE.EQ.6HAITKEN) GO TO 30                                   20SEPT    64
      CALL GMSET (IFNPARA,IFITPTR,NFP)                                  20SEPT    65
      KOMCOD = 7000000000000B + SHIFT(NFP,18)                           14APR76   63
      CALL GMSET(KOMCOD,IFITPTR,A)                                      BUILDFB   50
C                                                                       BUILDFB   51
```

```
      C         A SPLINE FIT IS NOT UNIQUELY DEFINED UNLESS THE POINTS ARRAYS  BUILDFB   52
      C         ARE KNOWN. ACCORDINGLY, IF THE FIT IS A SPLINE, STORE THE      BUILDFB   53
   60 C         X-COORDINATE ARRAY AND THE Y-COORDINATE ARRAY IN THE FIT BEAD  BUILDFB   54
      C         IN ADDITION TO THE ARRAY OF FIT PARAMETERS. OTHERWISE, STORE   BUILDFB   55
      C         ONLY THE ARRAY OF FIT PARAMETERS.                              BUILDFB   56
      C                                                                        BUILDFB   57
            IF(ITYPE .NE. 6HSPLINE) RETURN                                     BUILDFB   58
   65       C                                                                  BUILDFB   59
      C         STORE THE X-COORDINATE ARRAY.                                  BUILDFB   60
      C                                                                        BUILDFB   61
            KOMPCOD = KOMPCOD + NFP                                            BUILDFB   62
            CALL CMSET(KOMPCOD,IFITPTR,X)                                      BUILDFB   63
   70       C                                                                  BUILDFB   64
      C         STORE THE Y-COORDINATE ARRAY.                                  BUILDFB   65
      C                                                                        BUILDFB   66
            KOMPCOD = KOMPCOD + NFP                                            BUILDFB   67
            CALL CMSET(KOMPCOD,IFITPTR,Y)                                      BUILDFB   68
   75       RETURN                                                             BUILDFB   69
      C                                                                        20SEPT    66
      C         AITKEN FIT -- STORE NPOINTS, X AND Y ARRAYS                    20SEPT    67
      C                                                                        20SEPT    68
         30 CALL CMSET (IFNPARA,IFITPTR,NPOINTS)                               20SEPT    69
   80    CALL CMSET (IFNPARA+1,IFITPTR,NFP)                                    20SEPT    70
         KOMPCOD = 700000000078+SHIFT(NPOINTS,18)                              20SEPT    71
         CALL CMSET (KOMPCOD,IFITPTR,X)                                        20SEPT    72
         KOMPCOD=KOMPCOD+NPOINTS                                               20SEPT    73
         CALL CMSET (KOMPCOD,IFITPTR,Y)                                        20SEPT
   85    RETURN                                                                20SEPT
         END                                                                   BUILDFB   70
```

```
      SUBROUTINE SPLINC(N,XN,FN,GN,W,ISW,RR1,RRN)                        SPLINC     2
C                                                                        SPLINC     3
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *    SPLINC     4
C                                                                        SPLINC     5
C  CUBIC SPLINE INTERPOLATION ROUTINE.                                   SPLINC     6
C                                                                        SPLINC     7
C    N   = NUMBER OF DATA POINTS (I.E. THE NUMBER OF KNOTS)              SPLINC     8
C    XN  = ARRAY OF X-VALUES                                             SPLINC     9
C    FN  = ARRAY OF FUNCTION VALUES. (I.E. REQUIRED SPLINE VALUES)       SPLINC    10
C    GN  = ARRAY CONTAINING THE VALUES OF THE FIRST DERIVATIVE OF THE    SPLINC    11
C          SPLINE AT EACH KNOT. (RETURNED BY SPLINC)                     SPLINC    12
C    W   = A WORKING ARRAY DIMENSIONED 3*N-2                             SPLINC    13
C                                                                        SPLINC    14
C    ISW = 0 NO BOUNDARY CONDITIONS SPECIFIED                            SPLINC    15
C        = 1 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(1)          SPLINC    16
C        = 2 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(N)          SPLINC    17
C        = 3 USER SUPPLIES BOTH SECOND DERIVATIVE VALUES                 SPLINC    18
C        = 4 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(1)           SPLINC    19
C        = 5 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(N)           SPLINC    20
C        = 6 USER SUPPLIES BOTH FIRST DERIVATIVE VALUES                  SPLINC    21
C                                                                        SPLINC    22
C    RR1 = BOUNDARY VALUE AT XN(1)                                       SPLINC    23
C    RR2 = BOUNDARY VALUE AT XN(N)                                       SPLINC    24
C                                                                        SPLINC    25
C                                                                        SPLINC    26
C          CONDITIONS:                                                   SPLINC    27
C            1  N .GE. 3                                                 SPLINC    28
C            2  MUST HAVE XN(1) .LT. XN(2) .LT. ... .LT. XN(N)           SPLINC    29
C                                                                        SPLINC    30
C                                                                        SPLINC    31
C          THE RESULTS OF SPLINC CAN BE COUPLED WITH SPLIN2 TO EVALUATE  SPLINC    32
C          THE SPLINE OR ITS FIRST DERIVATIVE FOR ANY X WHICH SATISFIES  SPLINC    33
C             XN(1) .GE. X .LE. XN(N)                                    SPLINC    34
C                                                                        SPLINC    35
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *    SPLINC    36
C                                                                        SPLINC    37
C                                                                        SPLINC    38
      DIMENSION W(1),XN(1) ,FN(1),GN(1)                                  SPLINC    39
      IK=ISW+1                                                           SPLINC    40
      GO TO(1,2,3,4,60,70,80),IK                                         SPLINC    41
    1 R1=0.                                                              SPLINC    42
      RN=0.                                                              SPLINC    43
      GO TO 11                                                           SPLINC    44
    2 R1=RR1                                                             SPLINC    45
      RN=0.                                                              SPLINC    46
      GO TO 11                                                           SPLINC    47
    3 R1=0.                                                              SPLINC    48
      RN=RRN                                                             SPLINC    49
      GO TO 11                                                           SPLINC    50
    4 R1=RR1                                                             SPLINC    51
      RN=RRN                                                             SPLINC    52
   11 W(2)=1.0/(XN(2)-XN(1))                                             SPLINC    53
      W(1)=2.*W(2)                                                       SPLINC    54
      GN(1)=3.*(FN(2)-FN(1))*W(2)**2-R1/2.                               SPLINC    55
      K=2                                                                SPLINC    56
      J=2                                                                SPLINC    57
      W(K)=1.5/(XN(J)-XN(J-1))                                           SPLINC    58
```

```
      5 W(K+2)=1.0/(XN(J+1)-XN(J))                                                    SPLINC     59
        W(K+1)=2.*(W(K)+W(K+2))                                                       SPLINC     60
60      GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2                 SPLINC     61
        K=K+3                                                                         SPLINC     62
        J=J+1                                                                         SPLINC     63
        IF(J .EQ. N  ) GO TO 8                                                        SPLINC     64
        W(K)=W(K-1)                                                                   SPLINC     65
65      GO TO 5                                                                       SPLINC     66
      8 W(K)=1.0/(XN(N)-XN(N-1))                                                      SPLINC     67
        W(K+1)=2.*W(K)                                                                SPLINC     68
        GN(J)=3.*(FN(N)-FN(N-1))*W(K)**2+RN/2.                                        SPLINC     69
C                                                                                     SPLINC     70
70      MA=3*N-2                                                                      SPLINC     71
C                                                                                     SPLINC     72
        DO 10 J=4,MA,3                                                                SPLINC     73
     10 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                                              SPLINC     74
        DO 20 J=3,MA,3                                                                SPLINC     75
75   20 W(J)=W(J)/W(J-2)                                                              SPLINC     76
        IC=3                                                                          SPLINC     77
        DO 30 J=2,N                                                                   SPLINC     78
        GN(J)=GN(J)-W(IC)*GN(J-1)                                                     SPLINC     79
     30 IC=IC+3                                                                       SPLINC     80
80      N1=N+1                                                                        SPLINC     81
        IC=MA-2                                                                       SPLINC     82
        GN(N)=GN(N)/W(MA)                                                             SPLINC     83
        DO 40 J=2,N                                                                   SPLINC     84
        K=N1-J                                                                        SPLINC     85
85      GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                                           SPLINC     86
     40 IC=IC-3                                                                       SPLINC     87
C                                                                                     SPLINC     88
        RETURN                                                                        SPLINC     89
     60 H=XN(2)-XN(1)                                                                 SPLINC     90
90      W(2)=1./(XN(3)-XN(2))                                                         SPLINC     91
        W(1)=2.*(W(2)+1.0/H)                                                          SPLINC     92
        GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H                    SPLINC     93
        K=3                                                                           SPLINC     94
        J=2                                                                           SPLINC     95
95      W(K)=W(K-1)                                                                   SPLINC     96
     61 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                                  SPLINC     97
        W(K+1)=2.0*(W(K)+W(K+2))                                                      SPLINC     98
        GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2               SPLINC     99
        J=J+1                                                                         SPLINC    100
100     K=K+3                                                                         SPLINC    101
        IF(J .EQ. N-1) GO TO 62                                                       SPLINC    102
        W(K)=W(K-1)                                                                   SPLINC    103
        GO TO 61                                                                      SPLINC    104
     62 W(K)=XN(N)-XN(N-1)                                                            SPLINC    105
105     W(K+1)=2.*W(K)                                                                SPLINC    106
        GN(J)=3.*(FN(N)-FN(N-1))                                                      SPLINC    107
        NM=N-1                                                                        SPLINC    108
        IK=1                                                                          SPLINC    109
        GO TO 200                                                                     SPLINC    110
110 C                                                                                 SPLINC    111
     70 W(2)=1.0/(XN(2)-XN(1))                                                        SPLINC    112
        W(1)=2.*W(2)                                                                  SPLINC    113
        GN(1)=3.*(FN(2)-FN(1))*W(2)**2                                                SPLINC    114
        K=3                                                                           SPLINC    115
```

```
                      J=2                                                          SPLINC    116
                      W(K)=1.0/(XN(J)-XN(J-1))                                     SPLINC    117
                   71 W(K+2)=1.0/(XN(J+1)-XN(J))                                   SPLINC    118
                      W(K+1)=2.*(W(K)+W(K+2))                                      SPLINC    119
                      GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2 SPLINC   120
                      K=K+3                                                        SPLINC    121
                      J=J+1                                                        SPLINC    122
                      IF(J .EQ. N-1) GO TO 72                                      SPLINC    123
                      W(K)=W(K-1)                                                  SPLINC    124
                      GO TO 71                                                     SPLINC    125
                   72 W(K)=1.0/(XN(N-1)-XN(N-2))                                   SPLINC    126
                      W(K+2)=1.0/(XN(N)-XN(N-1))                                   SPLINC    127
                      W(K+1)=2.*(W(K)+W(K+2))                                      SPLINC    128
                      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2 SPLINC 129
                     1     -RRN*W(K+2)                                           SPLINC    130
                      NN=N-1                                                       SPLINC    131
                      IK=2                                                         SPLINC    132
                      GO TO 200                                                    SPLINC    133
                C                                                                  SPLINC    134
                   80 H=XN(2)-XN(1)                                                SPLINC    135
                      W(2)=1.0/(XN(3)-XN(2))                                       SPLINC    136
                      W(1)=2.0*(W(2)+1.0/H)                                        SPLINC    137
                      GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H   SPLINC    138
                      K=3                                                          SPLINC    139
                      J=2                                                          SPLINC    140
                      W(K)=W(K-1)                                                  SPLINC    141
                   81 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                 SPLINC    142
                      W(K+1)=2.0*(W(K)+W(K+2))                                     SPLINC    143
                      GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2 SPLINC 144
                      J=J+1                                                        SPLINC    145
                      K=K+3                                                        SPLINC    146
                      IF(J .EQ. N-2) GO TO 82                                      SPLINC    147
                      W(K)=W(K-1)                                                  SPLINC    148
                      GO TO 81                                                     SPLINC    149
                   82 W(K)=1.0/(XN(N-1)-XN(N-2))                                   SPLINC    150
                      W(K+2)=1.0/(XN(N)-XN(N-1))                                   SPLINC    151
                      W(K+1)=2.*(W(K)+W(K+2))                                      SPLINC    152
                      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2 SPLINC 153
                     1     -RRN*W(K+2)                                           SPLINC    154
                      NN=N-2                                                       SPLINC    155
                      IK=3                                                         SPLINC    156
                C                                                                  SPLINC    157
                  200 MA=3*NN-2                                                    SPLINC    158
                      DO 210 J=4,MA,3                                              SPLINC    159
                  210 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                             SPLINC    160
                      DO 220 J=3,MA,3                                              SPLINC    161
                  220 W(J)=W(J)/W(J-2)                                             SPLINC    162
                      IC=3                                                         SPLINC    163
                      DO 230 J=2,NN                                                SPLINC    164
                      GN(J)=GN(J)-W(IC)*GN(J-1)                                    SPLINC    165
                  230 IC=IC+3                                                      SPLINC    166
                      N1=NN+1                                                      SPLINC    167
                      IC=MA-2                                                      SPLINC    168
                      GN(NN)=GN(NN)/W(MA)                                          SPLINC    169
                      DO 240 J=2,NN                                                SPLINC    170
                      K=N1-J                                                       SPLINC    171
                      GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                          SPLINC    172
```

```
      240 IC=IC-3
C
      GO TO (300,320,340),IK
C
  300 N1=N+1
      DO 305 J=1,NN
  305 GN(N1-J)=GN(N1-J-1)
      GN(1)=RR1
      RETURN
  320 GN(N)=RRN
      RETURN
  340 DO 345 J=1,NN
  345 GN(N-J)=GN(N-J-1)
      GN(1)=RR1
      GN(N)=RRN
      RETURN
      END
```

```
SPLINC  173
SPLINC  174
SPLINC  175
SPLINC  176
SPLINC  177
SPLINC  178
SPLINC  179
SPLINC  180
SPLINC  181
SPLINC  182
SPLINC  183
SPLINC  184
SPLINC  185
SPLINC  186
SPLINC  187
SPLINC  188
SPLINC  189
```

```
      OVERLAY(7,0)                                                          PREFRST    2
      PROGRAM FRSTPLT                                                       PREFRST    3
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR     /BEADS/    2
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                      CYB777     1
     X     ISTR,IPUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                     6JULY77    3
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG              30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)        /DISKT/    3
      CALL GICLF(NCON)                                                      PREFRST    7
      CALL ORIGIN(-5.,5.)                                                   PREFRST    8
C     CALL DMSET(ICOM1+11,IPLTPTR,ICRV)                                     PREFRST    9
C     CALL DMSET(ICOM1+11,ICRV,IFIT)                                        PREFRST   10
C     IF(IFIT.NE.0)CALL AETSKC(7HFRSTPL2)                                   PREFRST   11
      CALL FREFIT                                                           PREFRST   21
      CALL AETSKC(7HFRSTPL2)                                                PREFRST   22
      END                                                                   PREFRST   23
```

```
      1                SUBROUTINE PREFIT                                                  PREFIT      2
                       COMMON/PREFIT/IVARS,FTYPE,IDEG,ITYSEL,IVSEL                        PREFIT      3
                       DIMENSION TYPE(3),IVAR(3)                                          PREFIT      4
                       COMMON/DISKT/I3D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG       1
      5          C     J=N*2**18  OMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)     /DISKT/     3
                       COMMON /GS/ I8UF(128),NCON,N8YTE,M8YTE,IGN,ISIN,                   CY6777      1
                      X     ISTR,I8UT,IMARK,ICODE,I8EAM,ISTYLE,IEOM,CPTO                 6JULY77     3
                       COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR  /BEADS/     2
                       COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                 /ITRIMT/    2
     10                COMMON/AXSBEAD/IAXSX1,IAXSXI,IAXSY1M,IAXSY1I,IAXSY2M,IAXSY2I,      /AXSBEAD/    2
                      X              IAXSY3M,IAXSY3I                                     /AXSBEAD/    3
                       COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/    2
                      X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/    3
                      X               IPFMN,IPFMV                                        /PLTBEAD/    4
     15                COMMON/SLFIT/ICR12,ICHNG                                           /SLFIT/     2
                       EXTERNAL PPLOT                                                     PREFIT      9
                       DATA TYPE/10HLEASTSQUAR,6HSPLINE,6HAITKEN/,IVAR/1HX,1HY,7HDEFAULT/ 23FEB       6
                       IF(ICHNG.EQ.0)GO TO 10                                             SLOPES     30
                       IVSEL=2                                                            SLOPES     31
     20                CALL OMGET(ICOM1+11,IPLTPTR,ICV3)                                  SLOPES     32
                       CALL OMGET(ICOM1+10,ICV3,IVARS)                                    SLOPES     33
                       IF(IVARS.EQ.1HY)GO TO 5                                            SLOPES     34
                       IVARS=1HX                                                          SLOPES     35
                       IVSEL=1                                                            SLOPES     36
     25          5     CONTINUE                                                           SLOPES     37
                       ITYSEL=2                                                           SLOPES     38
                       FTYPE=TYPE(ITYSEL)                                                 SLOPES
                       GO TO 50                                                           SLOPES
                10     CALL ANNOTA(0.,0.,25HPICK INDEPENDENT VARIABLE,25,IOD1,IGN,-0.-3,  PREFIT     13
     30                1-0)                                                               PREFIT     14
                       CALL ANNOTA (0.,-4.,1HX,1,IOD2,ISIN,1,-0,-0)                       3MARCH      40
                       CALL ANNOTA (2.,-4.,1HY,1,IOD3,ISIN,2,-0,-0)                       3MARCH      41
                       CALL ANNOTA (4.,-4.,7HDEFAULT,7,IOD4,ISIN,3,-0,-0)                 3MARCH      42
                       CALL ANNOTA (2.,-6.,6HACCEPT,6,IOD5,I8UT,4,-0,-0)                  3MARCH      43
     35          15    CALL GIBUT (0,NCON,ICOT,IVX)                                       3MARCH      44
                       IF (IVX.EQ.254) CALL STOP                                          3MARCH      45
                       IF (IVX.NE.4) GO TO 15                                             3MARCH      46
                       CALL GIFID (NCON,ICOT,IVSEL)                                       3MARCH      47
                       CALL GIBAS (IOD1,IOD2,IOD3,IOD4,IOD5)                              3MARCH      48
     40                IF(IVSEL.LT.1.OR.IVSEL.GT.3)GO TO 10                               PREFIT     20
                       IVARS=IVAR(IVSEL)                                                  PREFIT     21
                20     CALL ANNOTA(0.,0.,16HPICK TYPE OF FIT,16,IOD1,IGN,-0,-0,-0)        PREFIT     22
                       CALL ANNOTA(0.,-4.,12HLEAST SQUARE,12,IOD2,I8UT,1,-0,-0)           PREFIT     23
                       CALL ANNOTA (2.,-4.,6HSPLINE,6,IOD3,I8UT,2,-0,-0)                  23FEB       7
     45                CALL ANNOTA(4.,-4.,13HSTRAIGHT LINE,13,IOD4,I8UT,3,-0,-0)          PREFIT     25
                       CALL GIBUT(0,NCON,IOOT,ITYSEL)                                     PREFIT     26
                       CALL GIBAS(IOD1,IOD2,IOD3,IOD4)                                    PREFIT     27
                       IF(ITYSEL.LT.1.OR.ITYSEL.GT.3)GO TO 20                             PREFIT     28
                       FTYPE=TYPE(ITYSEL)                                                 PREFIT     29
     50                IF(ITYSEL.NE.1)GO TO 50                                            PREFIT     30
                30     CALL ANNOTA(0.,0.,42HSELECT DEGREE OF LEAST SQUARES POLYNOMIAL     PREFIT     31
                      142,IOD1,IGN,-0,-0,-0)                                              PREFIT     32
                       CALL ANNOTA(1.,-4.,1H1,1,IOD2,I8UT,1,-0,-0)                        PREFIT     33
                       CALL ANNOTA(2.,-4.,1H2,1,IOD3,I8UT,2,-0,-0)                        PREFIT     34
     55                CALL ANNOTA(3.,-4.,1H3,1,IOD4,I8UT,3,-0,-0)                        PREFIT     35
                       CALL ANNOTA(4.,-4.,1H4,1,IOD5,I8UT,4,-0,-0)                        PREFIT     36
                       CALL ANNOTA(5.,-4.,1H5,1,IOD6,I8UT,5,-0,-0)                        PREFIT     37
```

```
                CALL ANNOTA(6.,-4.,1H6,1,IDD7,IBUT,6,-0,-0)                    PREFIT     38
                CALL GIBUT(0,NCON,IDDT,IDEG)                                   PREFIT     39
60              CALL GIERAS(IDD1,IDD2,IDD3,IDD4,IDD5,IDD6,IDD7)                PREFIT     40
                IF(IDEG.LT.1.OR.IDEG.GT.6)GO TO 30                             PREFIT     41
             50 CONTINUE                                                       PREFIT     42
                CALL DMGET (IFAXISS,IFLTPTR,IAXSPTR)                           3MARCH     49
                CALL DMGET (IAXSXM,IAXSPTR,XMIN)                               3MARCH     50
65              CALL DMGET (IAXSXI,IAXSPTR,XINCR)                              3MARCH     51
                CALL DMGET (IAXSY1M,IAXSPTR,YMIN)                              3MARCH     52
                CALL DMGET (IAXSY1I,IAXSPTR,YINCR)                             3MARCH     53
                XMAX=XMIN+9.*XINCR                                             3MARCH     54
                YMAX=YMIN+6.*YINCR                                             3MARCH     55
70              IF (YMIN.LT.XMAX) GO TO 60                                     3MARCH     56
                TEMP=XMIN                                                      3MARCH     57
                XMIN=YMAX                                                      3MARCH     58
                XMAX=TEMP                                                      3MARCH     59
             60 IF (YMIN.LT.YMAX) GO TO 70                                     3MARCH     60
75              TEMP=YMIN                                                      3MARCH     61
                YMIN=YMAX                                                      3MARCH     62
                YMAX=TEMP                                                      3MARCH     63
             70 XINCR=ABS(XINCR)                                               3MARCH     64
                YINCR=ABS(YINCR)                                               3MARCH     65
80              CALL DMSET (IAXSXM,IAXSPTR,XMIN)                               3MARCH     66
                CALL DMSET (IAXSXI,IAXSPTR,XINCR)                              3MARCH     67
                CALL DMSET (IAXSY1M,IAXSPTR,YMIN)                              3MARCH     68
                CALL DMSET (IAXSY1I,IAXSPTR,YINCR)                             3MARCH     69
                CALL ITRIMR(IFLTPTR)                                           PREFIT     43
85              CALL SPLICE (IFLTPTR,PPLOT,12,1,INEXT,IPREV,0)                 14 JUNE    70
                END                                                            PREFIT     45
```

```
      SUBROUTINE PPLOT(IBEAD1,IBEAD,IVALU)                              PPLOT      2
      COMMON/DISKT/I3D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,I3DA,DATA)  STORES DATA(N) IN I8DA(1)    /DISKT/    3
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/  2
      EXTERNAL PCURVE2                                                  PPLOT      5
      CALL DMGET(ICOM1+8,IBEAD,IAX)                                     PPLOT      6
      CALL DMGET(ICOM1+2,IAX,XINCR)                                     PPLOT      7
      CALL DMGET(ICOM1+4,IAX,YINCR)                                     PPLOT      8
      CALL DMGET(ICOM1+11,IBEAD,ICRV1)                                  PPLOT      9
      CALL SPLICE2 (ICRV1,PCURVE2,2,0,1,2,0)                            PPLOT     10
      END                                                               PPLOT     11
```

```
      SUBROUTINE PCURVE2 (IBEAD1,IBEAD,IVALU)                                PCURVE2    2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/   2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/    3
      INTEGER FTYPE                                                          23FEB      8
      COMMON/PRESCFIT/IVARS,FTYPE,IDEG,ITYSEL,IVSEL                          PCURVE2    5
      DIMENSION W(742),DERIV(249),WORK(42),POLYCF(7)                         PCURVE2    6
      EQUIVALENCE (DERIV(1),POLYCF(1)),(W(249),WORK(1))                      PCURVE2    7
      IER=0                                                                  NEWFACE   31
      IV=IVARS                                                               PCURVE2    8
      CALL DMGET(ICOM1+6,IBEAD,NPOINTS)                                      23FEB      9
      CALL DMGET(ICOM1+12,IBEAD,ISTAT)                                       PCURVE2    9
      IF(ISTAT.EQ.6HINACTIVE)RETURN                                          PCURVE2   10
      CALL DMGET(ICOM1+10,IBEAD,IVOLD)                                       PCURVE2   11
      CALL DMGET(ICOM1+11,IBEAD,IFIT)                                        PCURVE2   12
      IF(IFIT.EQ.0)GO TO 20                                                  PCURVE2   13
      CALL DMGET(ICOM1+1,IFIT,OTYPE)                                         PCURVE2   14
      IF(FTYPE.NE.OTYPE)GO TO 9                                              PCURVE2   15
      IF(IVOLD.NE.IVARS)GO TO 9                                              PCURVE2   16
      IF(ITYSEL.NE.1)RETURN                                                  PCURVE2   17
      CALL DMGET(ICOM1+5,IFIT,NCOF)                                          PCURVE2   18
      IF(IDEG+1.EQ.NCOF)RETURN                                               PCURVE2   19
    9 CALL DMPURD(IFIT)                                                      PCURVE2   20
   20 CONTINUE                                                               PCURVE2   21
      IF(IVSEL.EQ.3)IV=IVOLD                                                 PCURVE2   22
      CALL SORTCK(IBEAD,IV,NPOINTS)                                          PCURVE2   23
      CALL DSCLEAN (NPOINTS)                                                 25JULY77
      IF(FTYPE.EQ.6HAITKEN)GO TO 295                                         23FEB
      IF(FTYPE.EQ.6HSPLINE)GO TO 200                                         PCURVE2   25
C     LEAST SQUARE                                                           PCURVE2   26
      NEP=IDEG+1                                                             PCURVE2   27
      IN=0                                                                   23FEB     11
      IF(IV.EQ.1HY)GO TO 100                                                 PCURVE2   28
      CALL FLSCF(X,Y,W,NPOINTS,IDEG,NMX,POLYCF,IN,XD,X0,WORK,IER)            23FEB     12
      IF (IER.NE.0) GO TO 350                                                3MARCH    70
      GO TO 300                                                              23FEB     13
  100 CALL FLSCF(Y,X,W,NPOINTS,IDEG,NMX,POLYCF,IN,XD,X0,WORK,IER)            23FEB     14
      IF (IER.NE.0) GO TO 350                                                3MARCH    71
      GO TO 300                                                              PCURVE2   32
  200 NEP=NPOINTS                                                            PCURVE2   33
      IF(IV.EQ.1HY)GO TO 205                                                 PCURVE2   34
      CALL SPLINC(NPOINTS,X,Y,DERIV,W,0,DUM1,DUM2)                           PCURVE2   35
      GO TO 300                                                              PCURVE2   36
  205 CALL SPLINC(NPOINTS,Y,X,DERIV,W,0,DUM1,DUM2)                           PCURVE2   37
      GO TO 300                                                              23FEB     15
  295 NEP=IDEG=1                                                             PCURVE2   38
  300 CONTINUE                                                               PCURVE2   39
      CALL BUILDF9(FTYPE,DERIV,NEP,IV,NPOINTS,ICRVFIT)                       PCURVE2   40
      CALL DMSET(ICOM1+11,IBEAD,ICRVFIT)                                     PCURVE2   41
      RETURN                                                                 3MARCH    72
C                                                                            3MARCH    73
C        ERROR - LEAST SQUARES                                               3MARCH    74
  350 CALL ERRORM (0.,-5.,58HERRORS IN LEAST SQUARES--POSSIBLE BAD INDEP     3MARCH    75
     1INDEPENDENT VARIABLE, 59,7HPCURVE2,6HINFORM)                           3MARCH    76
      RETURN                                                                 3MARCH    77
      END                                                                    PCURVE2   42
```

```
      1                 SUBROUTINE SPLICE(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)          SPLICE     2
                      C SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU   SPLICE     3
                      C IN WORD JMATCH OF EACH BEAD.                                    SPLICE     4
                      C WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND           SPLICE     5
      5               C THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                   SPLICE     6
                      C IF(IS.NE.1)BEAD IS NOT SPLICED OUT                              SPLICE     7
                      C POINTER TO PREVIOUS BEAD IN WORD JPREV                          SPLICE     8
                      C POINTER TO NEXT BEAD IN WORD JNEXT                              SPLICE     9
                        DATAICOMP/0100000000000/                                        SPLICE    10
     10                 NEXT=IBD1                                                       SPLICE    11
                        IPREV=IBD1                                                      SPLICE    12
                        IF(IBD1.NE.0)GOTO10                                             SPLICE    13
                        PRINT                                                           SPLICE    14
                      5 FORMAT(17H IBD1 IS NULL )                                       SPLICE    15
     15                 IS=2                                                            SPLICE    16
                        RETURN                                                          SPLICE    17
                     10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                           SPLICE    18
                        CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                             SPLICE    19
                        IF(ICKVALU.EQ.IVALU) GOTO15                                     SPLICE    20
     20                 IF(NEXTAD.EQ.0)GOTO12                                           SPLICE    21
                        IPREV=NEXT                                                      SPLICE    22
                        NEXT=NEXTAD                                                     SPLICE    23
                        GO TO 10                                                        SPLICE    24
                     12 WRITE (6,6) IVALU                                               CYB777    21
     25              6  FORMAT(19H NO MATCH FOR IVALU ,O20)                             SPLICE    26
                        IS=2                                                            SPLICE    27
                        RETURN                                                          SPLICE    28
                     15 CONTINUE                                                        SPLICE    29
                      C IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL       SPLICE    30
     30                 IF(NEXT.EQ.IBD1)GO TO 25                                        SPLICE    31
                      C OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE   SPLICE    32
                        IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                 SPLICE    33
                        IF(NEXTAD.EQ.0)GOTO22                                           SPLICE    34
                        CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                             SPLICE    35
     35                 IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                SPLICE    36
                     22 CONTINUE                                                        SPLICE    37
                        CALL SUBR(IBD1,NEXT,IVALU)                                      SPLICE    38
                        IF(IS.EQ.1)CALL DMRLBD(NEXT)                                    SPLICE    39
                        RETURN                                                          SPLICE    40
     40                 25 CALL SUBR(IBD1,IBD1,IVALU)                                   SPLICE    41
                        IF(NEXTAD.EQ.0)GO TO 40                                         SPLICE    42
                        NEXT=NEXTAD                                                     SPLICE    43
                     30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                             SPLICE    44
                        CALL SUBR(IBD1,NEXT,IVALU)                                      SPLICE    45
     45                 IF(IS.EQ.1)CALL DMRLBD(NEXT)                                    SPLICE    46
                        IF(NEXTAD.EQ.0 GO TO 40                                         SPLICE    47
                        NEXT=NEXTAD                                                     SPLICE    48
                        GO TO 30                                                        SPLICE    49
                     40 CONTINUE                                                        SPLICE    50
     50                 IF(IS.EQ.1)CALL DMRLAD(IBD1)                                    SPLICE    51
                        END                                                             SPLICE    52
```

```
      SUBROUTINE SORTCK(ICRVPTR,IV,NPOINTS)                                  SORTCK     2
C                                                                            SORTCK     3
C         THIS ROUTINE MOVES THE COORDINATE ARRAYS AND THE STATUS ARRAY      SORTCK     4
C         FROM THE DATA MANAGER FILE INTO CORE AND CHECKS TO SEE IF THE      SORTCK     5
C         POINTS HAVE TO BE SORTED. IF SO, THEN THEY ARE SORTED AND          SORTCK     6
C         MOVED BACK INTO THE DATA MANAGER FILE IN THEIR NEW SORT ORDER.     SORTCK     7
C                                                                            SORTCK     8
C         ICRVPTR - I - THE POINTER TO THE CURVE BEAD OF THE CURVE WHICH     SORTCK     9
C                   IS TO BE EXAMINED FOR SORT ORDER.                        SORTCK    10
C              IV - I - THE INDEPENDENT VARIABLE SUBMITTED BY THE USER.      SORTCK    11
C         NPOINTS - O - THE NUMBER OF POINTS IN THE COORDINATE BEADS.        SORTCK    12
C                                                                            SORTCK    13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,       /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                      /CRVBEAD/  3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/   2
C                                                                            SORTCK    16
C         MOVE THE COORDINATE ARRAYS AND THE STATUS ARRAY INTO CORE.         SORTCK    17
C                                                                            SORTCK    18
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                     SORTCK    19
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                      SORTCK    20
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                       SORTCK    21
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                       SORTCK    22
      KOMPCOD = 700000000018 + SHIFT(NPOINTS,18)                             SORTCK    23
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    24
      CALL DMGET(KOMPCOD,IXPTR,X)                                            SORTCK    25
      CALL DMGET(KOMPCOD,IYPTR,Y)                                            SORTCK    26
C                                                                            SORTCK    27
C         CHECK TO SEE IF THE POINTS NEED TO BE SORTED. THEY SHOULD BE       SORTCK    28
C         SORTED WITH THE SORT KEY EQUAL TO THE INDEPENDENT VARIABLE         SORTCK    29
C         SUBMITTED BY THE USER. IF THEY ARE ALREADY SORTED THAT WAY, NO     SORTCK    30
C         SORT IS NECESSARY. IF NOT, THEY MUST BE SORTED.                    SORTCK    31
C                                                                            SORTCK    32
      CALL DMGET(ICINDEP,ICRVPTR,IVAR)                                       SORTCK    33
      IF(IV.NE.IVAR) GO TO 10                                                SORTCK    34
      GO TO 20                                                               29APR     18
   10 IF(IV.EQ.IVAR) RETURN                                                  SORTCK    37
C                                                                            SORTCK    38
C         THE POINTS ARE NOT SORTED AS SPECIFIED BY THE USER. SORT           SORTCK    39
C         THEM AND DUPLICATE THE SORTED ARRAYS IN THE DATA MANAGER FILE.     SORTCK    40
C                                                                            SORTCK    41
   20 CALL SORTER (NPOINTS,IV)                                               29APR     19
      CALL DMSET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    43
      CALL DMSET(KOMPCOD,IXPTR,X)                                            SORTCK    44
      CALL DMSET(KOMPCOD,IYPTR,Y)                                            SORTCK    45
      CALL DMSET(ICINDEP,ICRVPTR,IV)                                         SORTCK    46
      RETURN                                                                 SORTCK    47
      END                                                                    SORTCK    48
```

```
      SUBROUTINE SORTER(NPOINTS,IV)                                     SORTER     2
C                                                                       SORTER     3
C         THIS ROUTINE SORTS THE COORDINATE ARRAYS AND THE STATUS ARRAY. SORTER     4
C         THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT        SORTER     5
C         VARIABLE.                                                     SORTER     6
C                                                                       SORTER     7
C         NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.  SORTER     8
C              IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE SORTER     9
C                       THREE ARRAYS (X,Y, AND NREC) ARE SORTED SO THAT THE SORTER    10
C                       X - COORDINATE ARRAY IS IN ASCENDING ORDER.   SORTER    11
C                       IF IV = 1HY, THEN THE THREE ARRAYS ARE SORTED SO SORTER    12
C                       THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER. SORTER 13
C                                                                       SORTER    14
      COMMON/POINTS/X(249),Y(249),NREC(248)                             /POINTS/   2
      DO 10 L=1,NPOINTS,L                                               SORTER    16
      M = 2 * L - 1                                                     SORTER    17
   10 CONTINUE                                                          SORTER    18
C                                                                       SORTER    19
C         START LOOP                                                    SORTER    20
C                                                                       SORTER    21
   20 M = M / 2                                                         SORTER    22
      IF(M .EQ. 0) GO TO 70                                             SORTER    23
      K = NPOINTS - M                                                   SORTER    24
      DO 60 J=1,K                                                       SORTER    25
      L = J                                                             SORTER    26
   30 IF(L .LT. 1) GO TO 60                                             SORTER    27
C                                                                       SORTER
C         USE THE SORT KEY.                                             SORTER
C                                                                       SORTER    30
      IF(IV .EQ. 1HY) GO TO 40                                          SORTER    31
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORTER    32
      GO TO 50                                                          SORTER    33
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORTER    34
C                                                                       SORTER    35
C         EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.               SORTER    36
C                                                                       SORTER    37
   50 TEMP = X(L)                                                       SORTER    38
      X(L) = X(L+M)                                                     SORTER    39
      X(L+M) = TEMP                                                     SORTER    40
C                                                                       SORTER    41
C         EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.               SORTER    42
C                                                                       SORTER    43
      TEMP = Y(L)                                                       SORTER    44
      Y(L) = Y(L+M)                                                     SORTER    45
      Y(L+M) = TEMP                                                     SORTER    46
C                                                                       SORTER    47
C         EXCHANGE POSITIONS IN THE STATUS ARRAY.                       SORTER    48
C                                                                       SORTER    49
      ITEMP = NREC(L)                                                   SORTER    50
      NREC(L) = NREC(L+M)                                               SORTER    51
      NREC(L+M) = ITEMP                                                 SORTER    52
      L=L-M                                                             8JUNE76   15
      GO TO 30                                                          SORTER    54
   60 CONTINUE                                                          SORTER    55
      GO TO 20                                                          SORTER    56
   70 RETURN                                                            SORTER    57
      END                                                               SORTER    58
```

```
      SUBROUTINE SPLINC(N,XN,FN,GN,W,ISW,RR1,RRN)                        SPLINC     2
C                                                                        SPLINC     3
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *    SPLINC     4
C                                                                        SPLINC     5
C  CUBIC SPLINE INTERPOLATION ROUTINE.                                   SPLINC     6
C                                                                        SPLINC     7
C    N   = NUMBER OF DATA POINTS (I.E. THE NUMBER OF KNOTS)              SPLINC     8
C    XN  = ARRAY OF X-VALUES                                             SPLINC     9
C    FN  = ARRAY OF FUNCTION VALUES. (I.E. REQUIRED SPLINE VALUES)       SPLINC    10
C    GN  = ARRAY CONTAINING THE VALUES OF THE FIRST DERIVATIVE OF THE    SPLINC    11
C          SPLINE AT EACH KNOT. (RETURNED BY SPLINC)                     SPLINC    12
C    W   = A WORKING ARRAY DIMENSIONED 3*N-2                             SPLINC    13
C                                                                        SPLINC    14
C    ISW = 0 NO BOUNDARY CONDITIONS SPECIFIED                            SPLINC    15
C        = 1 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(1)          SPLINC    16
C        = 2 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(N)          SPLINC    17
C        = 3 USER SUPPLIES BOTH SECOND DERIVATIVE VALUES                 SPLINC    18
C        = 4 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(1)           SPLINC    19
C        = 5 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(N)           SPLINC    20
C        = 6 USER SUPPLIES BOTH FIRST DERIVATIVE VALUES                  SPLINC    21
C                                                                        SPLINC    22
C    RR1 = BOUNDARY VALUE AT XN(1)                                       SPLINC    23
C    RR2 = BOUNDARY VALUE AT XN(N)                                       SPLINC    24
C                                                                        SPLINC    25
C                                                                        SPLINC    26
C          CONDITIONS:                                                   SPLINC    27
C            1  N .GE. 3                                                 SPLINC    28
C            2  MUST HAVE XN(1) .LT. XN(2) .LT. ... .LT. XN(N)           SPLINC    29
C                                                                        SPLINC    30
C                                                                        SPLINC    31
C          THE RESULTS OF SPLINC CAN BE COUPLED WITH SPLIN2 TO EVALUATE  SPLINC    32
C          THE SPLINE OR ITS FIRST DERIVATIVE FOR ANY X WHICH SATISFIES  SPLINC    33
C            XN(1) .GE. X .LE. XN(N)                                     SPLINC    34
C                                                                        SPLINC    35
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *    SPLINC    36
C                                                                        SPLINC    37
C                                                                        SPLINC    38
      DIMENSION W(1),XN(1) ,FN(1),GN(1)                                  SPLINC    39
      IK=ISW+1                                                           SPLINC    40
      GO TO(1,2,3,4,60,70,80),IK                                         SPLINC    41
    1 R1=0.                                                              SPLINC    42
      RN=0.                                                              SPLINC    43
      GO TO 11                                                           SPLINC    44
    2 R1=RR1                                                             SPLINC    45
      RN=0.                                                              SPLINC    46
      GO TO 11                                                           SPLINC    47
    3 R1=0.                                                              SPLINC    48
      RN=RRN                                                             SPLINC    49
      GO TO 11                                                           SPLINC    50
    4 R1=RR1                                                             SPLINC    51
      RN=RRN                                                             SPLINC    52
   11 W(2)=1.0/(XN(2)-XN(1))                                             SPLINC    53
      W(1)=2.*W(2)                                                       SPLINC    54
      GN(1)=3.*(FN(2)-FN(1))*W(2)**2-R1/2.                               SPLINC    55
      K=2                                                                SPLINC    56
      J=2                                                                SPLINC    57
      W(K)=1.0/(XN(J)-XN(J-1))                                           SPLINC    58
```

```
      5 W(K+2)=1.0/(XN(J+1)-XN(J))                                                    SPLINC     59
        W(K+1)=2.*(W(K)+W(K+2))                                                       SPLINC     60
   60   GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2                 SPLINC     61
        K=K+3                                                                         SPLINC     62
        J=J+1                                                                         SPLINC     63
        IF(J .EQ. N  ) GO TO 8                                                        SPLINC     64
        W(K)=W(K-1)                                                                   SPLINC     65
   65   GO TO 5                                                                       SPLINC     66
      8 W(K)=1.0/(XN(N)-XN(N-1))                                                      SPLINC     67
        W(K+1)=2.*W(K)                                                                SPLINC     68
        GN(J)=3.*(FN(N)-FN(N-1))*W(K)**2+RN/2.                                        SPLINC     69
C                                                                                     SPLINC     70
   70   MA=3*N-2                                                                      SPLINC     71
C                                                                                     SPLINC     72
        DO 10 J=4,MA,3                                                                SPLINC     73
     10 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                                              SPLINC     74
        DO 20 J=3,MA,3                                                                SPLINC     75
   75   20 W(J)=W(J)/W(J-2)                                                           SPLINC     76
        IC=3                                                                          SPLINC     77
        DO 30 J=2,N                                                                   SPLINC     78
        GN(J)=GN(J)-W(IC)*GN(J-1)                                                     SPLINC     79
     30 IC=IC+3                                                                       SPLINC     80
   80   N1=N+1                                                                        SPLINC     81
        IC=MA-2                                                                       SPLINC     82
        GN(N)=GN(N)/W(MA)                                                             SPLINC     83
        DO 40 J=2,N                                                                   SPLINC     84
        K=N1-J                                                                        SPLINC     85
   85   GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                                           SPLINC     86
     40 IC=IC-3                                                                       SPLINC     87
C                                                                                     SPLINC     88
        RETURN                                                                        SPLINC     89
     60 H=XN(2)-XN(1)                                                                 SPLINC     90
   90   W(2)=1./(XN(3)-XN(2))                                                         SPLINC     91
        W(1)=2.*(W(2)+1.0/H)                                                          SPLINC     92
        GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H                    SPLINC     93
        K=3                                                                           SPLINC     94
        J=2                                                                           SPLINC     95
   95   W(K)=W(K-1)                                                                   SPLINC     96
     61 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                                  SPLINC     97
        W(K+1)=2.0*(W(K)+W(K+2))                                                      SPLINC     98
        GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2               SPLINC     99
        J=J+1                                                                         SPLINC    100
  100   K=K+3                                                                         SPLINC    101
        IF(J .EQ. N-1) GO TO 62                                                       SPLINC    102
        W(K)=W(K-1)                                                                   SPLINC    103
        GO TO 61                                                                      SPLINC    104
     62 W(K)=XN(N)-XN(N-1)                                                            SPLINC    105
  105   W(K+1)=2.*W(K)                                                                SPLINC    106
        GN(J)=3.*(FN(N)-FN(N-1))                                                      SPLINC    107
        NN=N-1                                                                        SPLINC    108
        IK=1                                                                          SPLINC    109
        GO TO 200                                                                     SPLINC    110
  110 C                                                                               SPLINC    111
     70 W(2)=1.0/(XN(2)-XN(1))                                                        SPLINC    112
        W(1)=2.*W(2)                                                                  SPLINC    113
        GN(1)=3.*(FN(2)-FN(1))*W(2)**2                                                SPLINC    114
        K=3                                                                           SPLINC    115
```

```
      J=2                                                                SPLINC  116
      W(K)=1.0/(XN(J)-XN(J-1))                                           SPLINC  117
   71 W(K+2)=1.0/(XN(J+1)-XN(J))                                         SPLINC  118
      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC  119
      GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2      SPLINC  120
      K=K+3                                                              SPLINC  121
      J=J+1                                                              SPLINC  122
      IF(J .EQ. N-1) GO TO 72                                            SPLINC  123
      W(K)=W(K-1)                                                        SPLINC  124
      GO TO 71                                                           SPLINC  125
   72 W(K)=1.0/(XN(N-1)-XN(N-2))                                         SPLINC  126
      W(K+2)=1.0/(XN(N)-XN(N-1))                                         SPLINC  127
      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC  128
      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2    SPLINC  129
     1     -PRN*W(K+2)                                                   SPLINC  130
      NN=N-1                                                             SPLINC  131
      IK=2                                                               SPLINC  132
      GO TO 200                                                          SPLINC  133
C                                                                        SPLINC  134
   80 H=XN(2)-XN(1)                                                      SPLINC  135
      W(2)=1.0/(XN(3)-XN(2))                                             SPLINC  136
      W(1)=2.0*(W(2)+1.0/H)                                              SPLINC  137
      GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H         SPLINC  138
      K=3                                                                SPLINC  139
      J=2                                                                SPLINC  140
      W(K)=W(K-1)                                                        SPLINC  141
   81 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                       SPLINC  142
      W(K+1)=2.0*(W(K)+W(K+2))                                           SPLINC  143
      GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2    SPLINC  144
      J=J+1                                                              SPLINC  145
      K=K+3                                                              SPLINC  146
      IF(J .EQ. N-2) GO TO 82                                            SPLINC  147
      W(K)=W(K-1)                                                        SPLINC  148
      GO TO 81                                                           SPLINC  149
   82 W(K)=1.0/(XN(N-1)-XN(N-2))                                         SPLINC  150
      W(K+2)=1.0/(XN(N)-XN(N-1))                                         SPLINC  151
      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC  152
      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2    SPLINC  153
     1     -RRN*W(K+2)                                                   SPLINC  154
      NN=N-2                                                             SPLINC  155
      IK=3                                                               SPLINC  156
C                                                                        SPLINC  157
  200 MA=3*NN-2                                                          SPLINC  158
      DO 210 J=4,MA,3                                                    SPLINC  159
  210 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                                   SPLINC  160
      DO 220 J=3,MA,3                                                    SPLINC  161
  220 W(J)=W(J)/W(J-2)                                                   SPLINC  162
      IC=3                                                               SPLINC  163
      DO 230 J=2,NN                                                      SPLINC  164
      GN(J)=GN(J)-W(IC)*GN(J-1)                                          SPLINC  165
  230 IC=IC+3                                                            SPLINC  166
      N1=NN+1                                                            SPLINC  167
      IC=MA-2                                                            SPLINC  168
      GN(NN)=GN(NN)/W(MA)                                                SPLINC  169
      DO 240 J=2,NN                                                      SPLINC  170
      K=N1-J                                                             SPLINC  171
      GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                                SPLINC  172
```

```
      240 IC=IC-3                                                    SPLINC  173
C                                                                    SPLINC  174
      GO TO (300,320,340),IK                                         SPLINC  175
175   C                                                              SPLINC  176
      300 N1=N+1                                                     SPLINC  177
      DO 305 J=1,NN                                                  SPLINC  178
      305 GN(N1-J)=GN(N1-J-1)                                        SPLINC  179
      GN(1)=RR1                                                      SPLINC  180
180   RETURN                                                         SPLINC  181
      320 GN(N)=RRN                                                  SPLINC  182
      RETURN                                                         SPLINC  183
      340 DO 345 J=1,NN                                              SPLINC  184
      345 GN(N-J)=GN(N-J-1)                                          SPLINC  185
185   GN(1)=RR1                                                      SPLINC  186
      GN(N)=RRN                                                      SPLINC  187
      RETURN                                                         SPLINC  188
      END                                                            SPLINC  189
```

```
      1                 SUBROUTINE SPLICE2(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)          SPLICE2     2
                  C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU     SPLICE2     3
                  C     IN WORD JMATCH OF EACH BEAD.                                       SPLICE2     4
                  C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND              SPLICE2     5
      5           C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                     SPLICE2     6
                  C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                 SPLICE2     7
                  C     POINTER TO PREVIOUS BEAD IN WORD JPREV                             SPLICE2     8
                  C     POINTER TO NEXT BEAD IN WORD JNEXT                                 SPLICE2     9
                        DATA(OMP/0100000000008/                                          SPLICE2    10
     10                 NEXT=IBD1                                                          SPLICE2    11
                        IPREV=IBD1                                                         SPLICE2    12
                        IF(IBD1.NE.0)GOTO10                                                SPLICE2    13
                        WRITE (6,5)                                                      CYBZZZ    22
                    5   FORMAT (13H IBD1 IS NULL )                                         SPLICE2    15
     15                 IS=2                                                               SPLICE2    16
                        RETURN                                                             SPLICE2    17
                   10   CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                              SPLICE2    18
                        CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                SPLICE2    19
                        IF(ICKVALU.EQ.IVALU) GOTO15                                        SPLICE2    20
     20                 IF(NEXTAD.EQ.0)GOTO12                                              SPLICE2    21
                        IPREV=NEXT                                                         SPLICE2    22
                        NEXT=NEXTAD                                                        SPLICE2    23
                        GO TO 10                                                           SPLICE2    24
                   12   WRITE (6,6) IVALU                                                CYBZZZ    23
     25            6   FORMAT(19H NO MATCH FOR IVALU ,O20)                                SPLICE2    26
                        IS=2                                                               SPLICE2    27
                        RETURN                                                             SPLICE2    28
                   15   CONTINUE                                                           SPLICE2    29
                  C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL          SPLICE2    30
     30                 IF(NEXT.EQ.IBD1)GO TO 25                                           SPLICE2    31
                  C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE      SPLICE2    32
                        IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                    SPLICE2    33
                        IF(NEXTAD.EQ.0)GOTO22                                              SPLICE2    34
                        CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                SPLICE2    35
     35                 IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                   SPLICE2    36
                   22   CONTINUE                                                           SPLICE2    37
                        CALL SUBR(IBD1,NEXT,IVALU)                                         SPLICE2    38
                        IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    39
                        RETURN                                                             SPLICE2    40
     40                 25   CALL SUBR(IBD1,IBD1,IVALU)                                    SPLICE2    41
                        IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    42
                        NEXT=NEXTAD                                                        SPLICE2    43
                   30   CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                               SPLICE2    44
                        CALL SUBR(IBD1,NEXT,IVALU)                                         SPLICE2    45
     45                 IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    46
                        IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    47
                        NEXT=NEXTAD                                                        SPLICE2    48
                        GO TO 30                                                           SPLICE2    49
                   40   CONTINUE                                                           SPLICE2    50
     50                 IF(IS.EQ.1)CALL DMRLBC(IBD1)                                       SPLICE2    51
                        END                                                                SPLICE2    52
```

```
      SUBROUTINE BUILDF8(ITYPE,A,NFP,IV,NPOINTS,IFITPTR)                  BUILDF8    2
C                                                                         BUILDF8    3
C     THIS ROUTINE GETS A FIT BEAD AND FILLS IT WITH INFORMATION          BUILDF8    4
C     ABOUT THE FIT AND THE INTERVAL OVER WHICH IT APPLIES.               BUILDF8    5
C                                                                         BUILDF8    6
C       ITYPE - I - THE TYPE OF FIT; 10HLEASTSQUAR FOR A LEAST            BUILDF8    7
C               SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND               13SEPT    12
C               6HAITKEN FOR A STRAIGHT FIT.                              13SEPT    13
C           A - I - THE ARRAY OF FIT PARAMETERS; THE LEAST SQUARES        BUILDF8    9
C               POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT AND       BUILDF8   10
C               THE ARRAY OF DERIVATIVES FOR A SPLINE FIT.                BUILDF8   11
C               A DUMMY FOR THE STRAIGHT FIT.                             13SEPT    14
C         NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF           BUILDF8   12
C               COEFFICIENTS FOR A LEAST SQUARES FIT AND THE NUMBER       BUILDF8   13
C               OF POINTS FOR A SPLINE FIT.                               BUILDF8   14
C               NUMBER OF DEGREE OF FIT FOR STRAIGHT.                     13SEPT    15
C        IV - I - THE USER SUBMITTED INDEPENDENT VARIABLE.                BUILDF8   15
C     NPOINTS - I - THE NUMBER OF POINTS ON THE PLOT.                     BUILDF8   16
C     IFITPTR - O - THE POINTER TO THE FIT BEAD.                          BUILDF8   17
C                                                                         BUILDF8   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                      /AXSCALE/  2
      COMMON/DISPLAY/IDSTOP,IDDAD(90),NN,NNEND(10),IDSPEC(19),IDCF(41),   /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCD(23),IDMENU(20),           6JULY77    2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM  1SEPT      1
      COMMON/FITBEAD/IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA                14APR76    1
      COMMON/POINTS/X(248),Y(248),NREC(248)                               /POINTS/   2
C                                                                         BUILDF8
C     GET A FIT BEAD. THE LENGTH OF THE FIT                               BUILDF8
C     BEAD DEPENDS UPON THE TYPE OF FIT.                                  BUILDF8   25
C                                                                         BUILDF8   26
      IF(ITYPE .EQ. 10HLEASTSQUAR) CALL DMGTBD(NFP + 5,IFITPTR)           BUILDF8   27
      IF(ITYPE .EQ. 6HSPLINE) CALL DMGTBD(3 * NFP + 5,IFITPTR)            BUILDF8   28
      IF (ITYPE.EQ.6HAITKEN) CALL DMGTBD (2*NPOINTS+6,IFITPTR)            20SEPT    63
C                                                                         BUILDF8   29
C     STORE THE FIT TYPE.                                                 BUILDF8   30
C                                                                         BUILDF8   31
      CALL DMSET(IFTYPE,IFITPTR,ITYPE)                                    BUILDF8   32
C                                                                         BUILDF8   33
C     STORE THE INTERVAL AND INCREMENT INFORMATION.                       14APR76   62
C                                                                         BUILDF8   35
      IF(IV .EQ. 1HY) GO TO 10                                            BUILDF8   36
      CALL DMSET(IFIVVFP,IFITPTR,X(1))                                    BUILDF8   37
      CALL DMSET(IFIVVLP,IFITPTR,X(NPOINTS))                              BUILDF8   38
      CALL DMSET(IFIVI,IFITPTR,XINCR/20.)                                 BUILDF8   39
      GO TO 20                                                            BUILDF8   40
   10 CALL DMSET(IFIVVFP,IFITPTR,Y(1))                                    BUILDF8   41
      CALL DMSET(IFIVVLP,IFITPTR,Y(NPOINTS))                              BUILDF8   42
      CALL DMSET(IFIVI,IFITPTR,YINCR/20.)                                 BUILDF8   43
C                                                                         BUILDF8   44
C     STORE THE NUMBER OF FIT PARAMETERS                                  BUILDF8   45
C     AND THE ARRAY OF FIT PARAMETERS.                                    BUILDF8   46
C                                                                         BUILDF8   47
   20 IF (ITYPE.EQ.6HAITKEN) GO TO 30                                     20SEPT    64
      CALL DMSET (IFNPARA,IFITPTR,NFP)                                    20SEPT    65
      KOMPCOD = 700000000048 + SHIFT(NFP,18)                              14APR76   63
      CALL DMSET(KOMPCOD,IFITPTR,A)                                       BUILDF8   50
C                                                                         BUILDF8   51
```

```
      C          A SPLINE FIT IS NOT UNIQUELY DEFINED UNLESS THE POINTS ARRAYS  BUILDFB    52
      C          ARE KNOWN. ACCORDINGLY, IF THE FIT IS A SPLINE, STORE THE      BUILDFB    53
   60 C          X-COORDINATE ARRAY AND THE Y-COORDINATE ARRAY IN THE FIT BEAD  BUILDFB    54
      C          IN ADDITION TO THE ARRAY OF FIT PARAMETERS. OTHERWISE, STORE   BUILDFB    55
      C          ONLY THE ARRAY OF FIT PARAMETERS.                              BUILDFB    56
      C                                                                         BUILDFB    57
            IF(ITYPE .NE. 6HSPLINE) RETURN                                      BUILDFB    58
   65 C                                                                         BUILDFB    59
      C          STORE THE X-COORDINATE ARRAY.                                  BUILDFB    60
      C                                                                         BUILDFB    61
            KOMPCOD = KOMPCOD + NFP                                             BUILDFB    62
            CALL EMSET(KOMPCOD,IFITPTR,X)                                       BUILDFB    63
   70 C                                                                         BUILDFB    64
      C          STORE THE Y-COORDINATE ARRAY.                                  BUILDFB    65
      C                                                                         BUILDFB    66
            KOMPCOD = KOMPCOD + NFP                                             BUILDFB    67
            CALL EMSET(KOMPCOD,IFITPTR,Y)                                       BUILDFB    68
   75       RETURN                                                             BUILDFB    69
      C                                                                         20SEPT     66
      C          AITKEN FIT -- STORE NPOINTS, X AND Y ARRAYS                    20SEPT     67
      C                                                                         20SEPT     68
         30 CALL EMSET (IFNPARA,IFITPTR,NPOINTS)                                20SEPT     69
   80    CALL EMSET (IFNPARA+1,IFITPTR,NFP)                                     20SEPT     70
         KOMPCOD = 700000000078+SHIFT(NPOINTS,18)                               20SEPT     71
         CALL EMSET (KOMPCOD,IFITPTR,X)                                         20SEPT     72
         KOMPCOD=KOMPCOD+NPOINTS                                                20SEPT     73
         CALL EMSET (KOMPCOD,IFITPTR,Y)                                         20SEPT     74
   85    RETURN                                                                 20SEPT     75
         END                                                                    BUILDFB    70
```

```
      SUBROUTINE ITRIMR(IBEAD)                                          ITRIMR      2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/     2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG        1
C     J=N*2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/      3
      COMMON/OPTION/IOPTION,ISHINE                                      LASTBUX      1
      CALL DMGET(ICOM1+2,IBEAD,ITRIM)                                   ITRIMR       5
      IF(ITRIM.EQ.0)GO TO 10                                            ITRIMR       6
      IF (IOPTION.NE.4HTRIM) GO TO 10                                   3AUG        16
      INEXT=2                                                           ITRIMR       7
      IPREV=4                                                           ITRIMR       8
C                                                                       ITRIMR       9
C     TRIM MODE                                                         ITRIMR      10
      RETURN                                                            ITRIMR      11
   10 CONTINUE                                                          ITRIMR      12
C     CROSSPLOT MODE                                                    ITRIMR      13
      INEXT=1                                                           ITRIMR      14
      IPREV=3                                                           ITRIMR      15
      END                                                               ITRIMR      16
```

```
1                SUBROUTINE PRNTPLT(IPLTPTR)                                    PRNTPLT     2
                 COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                             /ITRIMT/    2
                 EXTERNAL SUBPLT                                                PRNTPLT     4
                 IF(IPLTPTR.EQ.0)RETURN                                         PRNTPLT     5
5                WRITE(N6,10)IPLTPTR                                            PRNTPLT     6
              10 FORMAT(1H1,50X,*LISTING OF P-M GROUP *,I10/)                   PRNTPLT     7
                 CALL ITRIMR(IPLTPTR)                                           PRNTPLT     8
                 CALL SPLICE(IPLTPTR,SUBPLT, 12,1,INEXT,IPREV,0)                5OCT      106
                 END                                                            PRNTPLT    10
```

```
      1                 SUBROUTINE SUBPLT(IBEAD1,IBEAD,IVALU)                                  PRNTPLT    13
                        COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG       1
                  C     J=N*2**15  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/     3
                        COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/    2
      5                 COMMON /TTEMP/TEMP(10)                                                 /TTEMP/     2
                        COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                     /ITRIMT/    2
                        EXTERNAL SUBCRV                                                        PRNTPLT    18
                        JJ=14*2**18                                                            PRNTPLT    19
                        CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                        PRNTPLT    20
     10                 WRITE(N6,10)IBEAD                                                      PRNTPLT    21
                   10   FORMAT(1HO,2X ,*LISTING OF PLOT *,I10)                                 PRNTPLT    22
                        WRITE(N6,20)(NREC(I),I=1,14)                                           PRNTPLT    23
                   20   FORMAT(1H ,12I8,1X,A10,1X,F10.5)                                       PRNTPLT    24
                        ITITLE=NREC(6)                                                         PRNTPLT    25
     15                 CALL PRNTEXT(ITITLE,10H      TITLE)                                    PRNTPLT    26
                        JJ=5*2**18                                                             PRNTPLT    27
                        IF(NREC(5).EQ.0)GO TO 45                                               PRNTPLT    28
                        CALL DMGET(ICOM7+JJ,NREC(5),TEMP)                                      PRNTPLT    29
                        WRITE(N6,30)NREC(5)                                                    PRNTPLT    30
     20                 30   FORMAT(1HO,6X ,*LISTING OF PLOT GROUP *,I10)                      PRNTPLT    31
                        WRITE(N6,40)(TEMP(I),I=1,7)                                            PRNTPLT    32
                   40   FORMAT(1H ,7I8)                                                        PRNTPLT    33
                        CALL PRNTEXT(TEMP(5),10H CONDITION)                                    PRNTPLT    34
                   45   CONTINUE                                                               PRNTPLT    35
     25                 IF(NREC(7).EQ.0)GO TO 65                                               PRNTPLT    36
                        WRITE(N6,50)NREC(7)                                                    PRNTPLT    37
                   50   FORMAT(1HO,6X ,*LISTING OF AXIS *,I10)                                 PRNTPLT    38
                        CALL DMGET(ICOM7+JJ,NREC(7),TEMP)                                      PRNTPLT    39
                        WRITE(N6,60)(TEMP(I),I=1,7)                                            PRNTPLT    40
     30                 60   FORMAT(1H ,4I8,1X,A10,I5,1X,A10)                                  PRNTPLT    41
                        CALL PRNTEXT(TEMP(3),10H       AXIS)                                   PRNTPLT    42
                   65   CONTINUE                                                               PRNTPLT    43
                        IF(NREC(9).EQ.0)GO TO 75                                               PRNTPLT    44
                        WRITE(N6,70)NREC(9)                                                    PRNTPLT    45
     35                 70   FORMAT(1HO,6X ,*LISTING OF P-M *,I10)                             PRNTPLT    46
                        CALL DMGET(ICOM7+JJ,NREC(9),TEMP)                                      PRNTPLT    47
                        WRITE(N6,40)(TEMP(I),I=1,7)                                            PRNTPLT    48
                        PARAM = TEMP(6)                                                        PRNTPLT    49
                        TMULT = TEMP(7)                                                        PRNTPLT    50
     40                 CALL PRNTEXT(PARAM,10H PARAMETER)                                      PRNTPLT    51
                        CALL PRNTEXT(TMULT,10HMULTIPLIER)                                      PRNTPLT    52
                   75   CONTINUE                                                               PRNTPLT    53
                        IF(NREC(10).EQ.0)GO TO 95                                              PRNTPLT    54
                        WRITE(N6,80)NREC(9)                                                    PRNTPLT    55
     45                 80   FORMAT(1HO,6X ,*LISTING OF AXIS SCALE *,I10)                      PRNTPLT    56
                        JJ=8*2**18                                                             T30AUG     23
                        CALL DMGET(ICOM7+JJ,NREC(8),TEMP)                                      PRNTPLT    58
                        WRITE (N6,90) (TEMP(I),I=1,8)                                          T30AUG     24
                   90   FORMAT(1H ,5X,4G16.8)                                                  22JUNE     83
     50                 95   CONTINUE                                                          PRNTPLT    61
                        IF(NREC(11).EQ.0)GO TO 99                                              PRNTPLT    62
                        CALL SPLICS2(NREC(11),SUBCRV,2,0,1,2,0)                                PRNTPLT    63
                   99   CONTINUE                                                               PRNTPLT    64
                        END                                                                    PRNTPLT    65
```

```
      1                SUBROUTINE SUBCRV(IBEAD1,IBEAD,IVALU)                                  PRNTPLT    68
                       COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                     /ITRIMT/    2
                       COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/    2
                       COMMON /TTEMP/TEMP(10)                                                 /TTEMP/     2
      5                COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG       1
                C      J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/     3
                       WRITE(N6,10)IBEAD                                                      PRNTPLT    73
                 10    FORMAT(1H0,11X,*LISTING OF CURVE *,I10)                                PRNTPLT    74
                       JJ=12*2**18                                                            PRNTPLT    75
      10               CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                        PRNTPLT    76
                       IF(NREC(1).NE.0)GO TO 15                                               PRNTPLT    77
                       WRITE(N6,19)(NREC(I),I=1,12)                                           PRNTPLT    78
                 19    FORMAT(1H ,11X,2I8,1X,A10,1X,G16.8,I5,I3,3I8,1X,I2 ,1X,I8,1X,I10)      22JUNE     84
                       GO TO 16                                                               PRNTPLT    80
      15               WRITE(N6,20)(NREC(I),I=1,12)                                           PRNTPLT    81
                 16    CONTINUE                                                               PRNTPLT    82
                 20    FORMAT(1H ,11X,2I8,1X,A10,1X,G16.8,I5,I3,3I8,1X,A2 ,1X,I8,1X,A10)      22JUNE     85
                       IR=NREC(7)                                                             PRNTPLT    84
                       IY=NREC(8)                                                             PRNTPLT    85
      20               IY=NREC(9)                                                             PRNTPLT    86
                       NPTS=NREC(5)                                                           PRNTPLT    87
                       IFIT=NREC(11)                                                          PRNTPLT    88
                       JJ=NPTS*2**18                                                          PRNTPLT    89
                       IF(IX.EQ.0)GO TO 35                                                    PRNTPLT    90
      25               CALL DMGET(ICOM7+JJ,IR,NREC)                                           PRNTPLT    91
                       CALL DMGET(ICOM7+JJ,IX,X)                                              PRNTPLT    92
                       CALL DMGET(ICOM7+JJ,IY,Y)                                              PRNTPLT    93
                       WRITE(N6,30)                                                           PRNTPLT    94
                 30    FORMAT(1H0,18X,*REC*,17X,1HX,19X,1HY)                                  22JUNE     86
      30               WRITE(N6,40)(NREC(I),X(I),Y(I),I=1,NPTS)                               PRNTPLT    96
                 40    FORMAT(1H ,11X,I10,10X,G16.8,4X,G16.8)                                 22JUNE     87
                 35    CONTINUE                                                               PRNTPLT    98
                       IF(IFIT.LE.0)RETURN                                                    PRNTPLT    99
                       CALL DMGET(ICOM1+1,IFIT,FTYPE)                                         PRNTPLT   100
      35               CALL DMGET(ICOM1+2,IFIT,XO)                                            PRNTPLT   101
                       CALL DMGET(ICOM1+3,IFIT,XN)                                            PRNTPLT   102
                       CALL DMGET(ICOM1+4,IFIT,DELX)                                          PRNTPLT   103
                       IF(FTYPE.EQ.6HAITKEN)GO TO 205                                         TOM        29
                       CALL DMGET(ICOM1+5,IFIT,NP)                                            PRNTPLT   104
      40               JJ=NP*2**18+5                                                          PRNTPLT   105
                       CALL DMGET(ICOM7+JJ,IFIT,X)                                            PRNTPLT   106
                       WRITE(N6,100)FTYPE,XO,XN,DELX                                          PRNTPLT   107
                100    FORMAT(1H0,15X,A10,* FIT *,3F16.5)                                     PRNTPLT   108
                       WRITE(N6,101)NP,(X(I),I=1,NP)                                          PRNTPLT   109
      45        101    FORMAT(1H0,20X,I5,* PARAMETERS ARE */(30X,5E16.8))                     PRNTPLT   110
                       RETURN                                                                 TOM        30
                205    CALL DMGET(ICOM1+6,IFIT,IDEG)                                          TOM        31
                       WRITE (N6,100) FTYPE,XO,XN,DELX                                        27MAY       8
                       WRITE(N6,105)IDEG                                                      TOM        33
      50        105    FORMAT(1H ,15X,*DEGREE  *,I2)                                          TOM        34
                       END                                                                    PRNTPLT   111
```

```
      SUBROUTINE PRNTEXT(IBEAD,TNAME)                                    PRNTEXT    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/    3
      COMMON /TTEMP/TEMP(10)                                             /TTEMP/    2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                 /ITRIMT/   2
      INTEGER TEMP                                                       PRNTEXT    6
      IBD=IBEAD                                                          PRNTEXT    7
      JJ=10*2**18                                                        PRNTEXT    8
   99 IF(IBD.LE.0)RETURN                                                 PRNTEXT    9
      CALL DMGET(ICOM7+JJ,IBD,TEMP)                                      PRNTEXT   10
      NW=TEMP(2)/10                                                      PRNTEXT   11
      IF(( TEMP(2)-10*NW).GT.0)NW=NW+1                                   PRNTEXT   12
      NW=NW+2                                                            PRNTEXT   13
      WRITE(N6,100)TNAME,IBD,(TEMP(I),I=1,NW)                           PRNTEXT   14
  100 FORMAT(1HC,30X,A10,* TEXT *,I10/1H ,15X,2I10,1X,8A10)              PRNTEXT   15
      IBD=TEMP(1)                                                        PRNTEXT   16
      GO TO 99                                                           PRNTEXT   17
      END                                                                PRNTEXT   18
```

```
      SUBROUTINE MRCLEAN(NPOINTS)                                          MRCLEAN     2
C                                                                          MRCLEAN     3
C         THIS ROUTINE CLEANS UP THE CORE RESIDENT POINTS ARRAYS BY        MRCLEAN     4
C         REMOVING INACTIVE POINTS AND POINTS WHICH ARE OFF SCALE FROM     MRCLEAN     5
C         THE IN-CORE POINTS ARRAYS. PRIMARY INPUTS TO THE ROUTINE ARE     MRCLEAN     6
C         THE POINTS ARRAYS, THE STATUS ARRAY, AND THE SCALE INFORMATION   MRCLEAN     7
C         THE PRIMARY OUTPUT FROM THE ROUTINE IS SANITIZED POINTS ARRAYS   MRCLEAN     8
C         WHICH ARE READY TO BE FED TO THE CURVE FITTING ROUTINES.         MRCLEAN     9
C         NOTE---THE DATA MANAGER FILE CONTINUES TO HOLD THE               MRCLEAN    10
C               UNSANITIZED ARRAYS.                                        MRCLEAN    11
C                                                                          MRCLEAN    12
C         NPOINTS - I/O - WHEN NPOINTS IS INPUT, IT IS EQUAL TO THE        MRCLEAN    13
C                   NUMBER OF POINTS IN THE COORDINATE BEADS. WHEN         MRCLEAN    14
C                   NPOINTS IS OUTPUT, IT IS EQUAL TO THE NUMBER OF        MRCLEAN    15
C                   POINTS WHICH ARE DISPLAYED ON THE SCREEN. THE          MRCLEAN    16
C                   TWO ARE NOT NECESSARILY THE SAME.                      MRCLEAN    17
C                                                                          MRCLEAN    18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                       /AXSCALE/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                /POINTS/    2
C                                                                          MRCLEAN    21
C         NPNOP --- THE NUMBER OF POINTS NOT ON THE PLOT.                  MRCLEAN    22
C                                                                          MRCLEAN    23
      NPNOP = 0                                                            MRCLEAN    24
      DO 20 I=1,NPOINTS                                                    MRCLEAN    25
      IF(NREC(I) .LT. 0) GO TO 10                                          MRCLEAN    26
      IF( (X(I) .LT. XMIN) .OR. (X(I) .GT. XMAX) ) GO TO 10                MRCLEAN    27
      IF( (Y(I) .LT. YMIN) .OR. (Y(I) .GT. YMAX) ) GO TO 10                MRCLEAN    28
C                                                                          MRCLEAN    29
C         THE "I-TH" POINT IS ON THE PLOT(CURRENTLY DISPLAYED ON THE       MRCLEAN    30
C         SCREEN). IF ANY POINTS PRIOR TO THE "I-TH" POINT ARE NOT ON      MRCLEAN    31
C         THE PLOT, THEN MOVE THE "I-TH" POINT INTO THE GAP CREATED        MRCLEAN    32
C         BY THE MISSING POINT(S). OTHERWISE, GO TO THE "(I+1)-ST" POINT   MRCLEAN    33
C                                                                          MRCLEAN    34
      IF(NPNOP .EQ. 0) GO TO 20                                            MRCLEAN    35
      X(I-NPNOP) = X(I)                                                    MRCLEAN    36
      Y(I-NPNOP) = Y(I)                                                    MRCLEAN    37
      GO TO 20                                                             MRCLEAN    38
C                                                                          MRCLEAN    39
C         THE POINT IS NOT ON THE PLOT.                                    MRCLEAN    40
C                                                                          MRCLEAN    41
   10 NPNOP = NPNOP + 1                                                    MRCLEAN    42
   20 CONTINUE                                                             MRCLEAN    43
      NPOINTS = NPOINTS - NPNOP                                            MRCLEAN    44
      RETURN                                                               MRCLEAN    45
      END                                                                  MRCLEAN    46
```

```
      OVERLAY(11,0)                                                      TABLE     2
      PROGRAM TABLE                                                      TABLE     3
C      CONSTRUCTS TRIM TABLES FOR PLOTS STORED AT IPLTPTR                TABLE     4
      COMMON /BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/   2
      COMMON/DISKT/IBD1,ICOM1,ICOM2,XVAL,XX(20),YY(20),IC,NDEG           30AUG     1
C     J=N*2**18  CMSET(ICOM2+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/   3
      DIMENSION CLNAME(4),CLVALU(4)                                      1SEPT     9
      EXTERNAL TABRL                                                     TABLE     7
      COMMON/CLINCR/CLMIN,DELCL                                          30JULY76  3
      DATA NCL/10/,CLNAME/6HCL MIN,9HDELTA CL,6HCL MAX,6HDEGREE/         1SEPT     10
     1,CLVALU/-.3,.1,1.1,1./                                             1SEPT     11
      CALL IWPLGR (IPLTPTR,2,4,IONE)                                     FIX2      1
      CALL VARIN(CLNAME,CLVALU ,4,-100,1000,YY)                          1SEPT     12
C     INPUT CLMIN,DELCL,CLMAX                                            30JULY76  6
      CLMIN=CLVALU(1)                                                    1SEPT     13
      DELCL=CLVALU(2)                                                    1SEPT     14
      CLMAX=CLVALU (3)                                                   1SEPT     15
      NDEG=CLVALU(4)                                                     1SEPT     16
      NCL=(CLMAX-CLMIN)/DELCL+1.5                                        30JULY76  10
      IF (NCL.GT.20) NCL=20                                              30JULY76  11
      CALL CMGET(ICOM1+2,IPLTPTR,NEXTT)                                  TABLE     10
      IF(NEXTT.NE.0)GO TO 10                                             TABLE     11
      CALL ERROFM(-4.,8.,30HTHESE PLOTS CANNOT BE TRIMMED.,30,           TABLE     12
     1  10HTRIM TABLE,6HINFORM)                                          TABLE     13
      CALL AETSKC (7HQTRBACK)                                            16NOV     14
   10 CONTINUE                                                           TABLE     16
C     IF PRIOR TABLE STORED, RELEASE IT                                  TABLE     17
      IF(IBD1.EQ.0) GO TO 20                                             TABLE     18
      CALL SPLICE(IBD1,TABRL,10,IBD1,9,8,1)                              TABLE     19
   20 CALL STOTABL(IPLTPTR,NCL)                                          TABLE     20
C                                                                        TABLE     21
C      PRINT TRIM TABLES                                                 TABLE     22
C                                                                        TABLE     23
      IBTRM=IBD1                                                         TABLE     24
      CALL PRNTRIM(IBTRM)                                                TABLE     25
C     CALL AETSKC(7HQTRBACK)                                             22JUNE    3
      CALL AETSKC(5HQTRIM)                                               22JUNE    4
      END                                                                TABLE     27
```

```
      SUBROUTINE STOTA9L(IPLTPTR,NCL)                                   STOTA9L    2
      COMMON/TTABLE/CL(20),DELC(20),ALPHA(20),CO(20)                    /TTABLE/   2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          3CAUG      1
C     J=M*2**15  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)    /DISKT/    3
      IP1=IPLTPTR                                                       STOTA9L    5
    5 IF(IP1.EQ.0)RETURN                                                STOTA9L    6
      CALL DMGET(ICOM1+11,IP1,ICRV1)                                    STOTA9L    7
      CALL DMGET(ICOM1+2,IP1,IP2)                                       STOTA9L    8
      CALL DMGET(ICOM1+11,IP2,ICRV2)                                    STOTA9L    9
      CALL DMGET(ICOM1+2,IP2,IP3)                                       STOTA9L   10
      CALL DMGET(ICOM1+11,IP3,ICRV3)                                    STOTA9L   11
      CALL ETABLE(ICRV1,ICRV2,ICRV3,NCL,NDEG)                           STOTA9L   12
      CALL SAVET(IP3,NCL)                                               STOTA9L   13
      CALL DMGET(ICOM1+2,IP3,IP1)                                       STOTA9L   14
      GO TO 5                                                           STOTA9L   15
      END                                                               STOTA9L   16
```

```
1                SUBROUTINE PRNTRM(IBTRM)                                PRNTRM     2
                 C     PRINTS TRIM TABLES                                PRNTRM     3
                       EXTERNAL TPRNT                                    PRNTRM     4
                       WRITE (6,99)                                      CY8777    12
5                99    FORMAT(1H1,50X,*TRIM TABLES*)                     22JUNE     6
                       CALL SPLICE(IBTRM,TPRNT,10,IBTRM,9,8,0)           PRNTRM     5
                       END                                               PRNTRM     6
```

```
1                SUBROUTINE TPRNT(I881,IBEAD,IVALU)                                  PRNTRM     7
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG            30AUG      1
           C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)      /DISKT/    3
                 COMMON/POINTS/X(248),Y(248),NREC(248)                               /POINTS/   2
5                DIMENSION IBD(10),ANAME(4)                                          PRNTRM    10
                 DATA ANAME/2HCL,4HDELC,5HALPHA,2HCD/                                22JUNE     7
                 J=10*2**18                                                          PRNTRM    12
                 CALL DMGET(ICOM7+J,IBEAD,IBD)                                       PRNTRM    13
                 WRITE (6,100) (IBD(I),I=1,10)                                       CY8777    13
10           100 FORMAT(1HC,5I10,G16.8,1X,A10,3I10)                                  22JUNE     8
                 NPTS=IBD(5)                                                         PRNTRM    16
                 J=NPTS*2**18                                                        PRNTRM    17
                 DO 20 II=1,4                                                        PRNTRM    18
                 CALL DMGET(ICOM7+J,IBD(II),X)                                       PRNTRM    19
15               WRITE (6,199) ANAME(II)                                             CY8777    14
           199   FORMAT(1H ,A10)                                                     PRNTRM    21
                 WRITE (6,200) (X(I),I=1,NPTS)                                       CY8777    15
           200   FORMAT(1H ,10G13.5)                                                 PRNTRM    23
           20    CONTINUE                                                            PRNTRM    24
20               END                                                                 PRNTRM    25
```

```
      SUBROUTINE FTABLE(ICRV1,ICRV2,ICRV3,NCL,JDEG)                          FTABLE     2
      EXTERNAL PCURVE                                                        FTABLE     3
      COMMON/DISKT/I9D1,ICOM1,ICOM2,XVAL,XX(20),YY(20),IC,NDEG               30AUG      1
C     J=P*2**18  OFSET(ICOM2+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)         /DISKT/    3
      COMMON/PNVAL/PPNAM,PPVAL                                               2NOV      59
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/   2
      COMMON/TTABLE/CL(20),DELC(20),ALPHA(20),CO(20)                         /TTABLE/   2
      COMMON /TTEMP/TEMP(10)                                                 /TTEMP/    2
      COMMON /GP/ I8UF(128),NCON,MBYTE,MBYTE,IGN,ISIN,                       CY8777     1
     X     ISTR,I8UT,IMARK,ICODE,I8BAM,ISTYLE,IEOM,CPTO                      6JULY77    3
C     ITEMFL=1  MEANS LOOK AT INTERMEDIATE TRIM PLOTS                        2NOV      62
      COMMON/CLINCR/CLMIN,DELCL                                              30JULY76  12
      DIMENSION GN(20),W(60)                                                 30JULY76  13
      DATA ISTOP/1/                                                          22JUNE     9
      DO 10 K=1,NCL                                                          30JULY76  14
      IV=1HY                                                                 30JULY76  15
      CL(K)=CLMIN+(K-1)*DELCL                                                30JULY76  16
      IC=0                                                                   30JULY76  17
      XVAL=CL(K)                                                             30JULY76  18
      PPNAM=2HCL                                                             2NOV      74
      PPVAL=XVAL                                                             2NOV      75
      CALL SPLICE (ICRV1,PCURVE,2,0,1,2,0)                                   30JULY76  19
C     XX=CM, YY=DEL                                                          30JULY76  20
      CALL SORT2 (XX,YY,IC,1HX)                                              30JULY76  21
      JDEG=2                                                                 30JULY76  22
      CALL INTERP (XX,YY,IC,JDEG,0.,DELC(K),TEMP,IER)                        30JULY76  23
      IF(IER.LE.1)GO TO 10                                                   22JUNE
      WRITE (6,999) K,IC,JDEG,IER,NCL,DELCL,CL(K),DELC(K)                    CY8777
  999 FORMAT(* FTABLE*,5I5,3G16.8)                                           22JUNE    12
      WRITE (6,1000) (XX(I),YY(I),I=1,IC)                                    CY8777    17
 1000 FORMAT(2G20.8)                                                         22JUNE    14
      IF(ISTOP.NE.1)STOP 22                                                  22JUNE    15
   10 CONTINUE                                                               FTABLE    22
      IV=1HY                                                                 FTABLE    23
      DO 20 K=1,NCL                                                          FTABLE    24
      IC=0                                                                   FTABLE    25
      XVAL=CL(K)                                                             FTABLE    26
      PPNAM=2HCL                                                             2NOV      88
      PPVAL=XVAL                                                             2NOV      89
      CALL SPLICE(ICRV2,PCURVE,2,0,1,2,0)                                    FTABLE    27
C     XX=ALPHA,YY=DEL                                                        FTABLE    28
      CALL SORT2(XX,YY,IC,1HY)                                               FTABLE    29
      CALL INTERP(YY,XX,IC,JDEG,DELC(K),ALPHA(K),TEMP,IER)                   FTABLE    30
      IF(ISTOP.NE.1)STOP 33                                                  22JUNE    24
   20 CONTINUE                                                               FTABLE    32
      IV=1HY                                                                 FTABLE    33
      DO 30 K=1,NCL                                                          FTABLE    34
      IC=0                                                                   FTABLE    35
      XVAL=ALPHA(K)                                                          FTABLE    36
      PPNAM=2HALPHA                                                          2NOV      102
      PPVAL=XVAL                                                             2NOV      103
      CALL SPLICE(ICRV3,PCURVE,2,0,1,2,0)                                    FTABLE    37
C     XX=CD,YY=DEL                                                           FTABLE    38
      CALL SORT2(XX,YY,IC,1HY)                                               FTABLE    39
      CALL INTERP(YY,XX,IC,JDEG,DELC(K),CD(K),TEMP,IER)                      FTABLE    40
      IF(IER.LE.1)GO TO 30                                                   22JUNE    25
      IF(ISTOP.NE.1)STOP 44                                                  22JUNE    26
```

```
      30  CONTINUE                                                  FTABLE     42
          END                                                       FTABLE     43
```

```
      SUBROUTINE TEMP80(XIN,YIN,NPTS,IV,ISYM,XNAME,YNAME,ITEM80)          TEMP80    2
      COMMON/TDIS/BB(4),ASCL(8)                                          TEMP80    3
      COMMON/PNVAL/PPNAM,PPVAL                                           2NOV    107
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NOEG           30AUG     1
C     J=N*2**18  DMSET(ICOM7+J,I80A,DATA)  STORES DATA(N) IN I80A(1)     /DISKT/   3
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/  2
      CALL DMGT80(14,ITEM80)                                             TEMP80    6
      CALL DMGT80(12,ICRV)                                               TEMP80    7
      CALL DMSET(ICOM1+3,ICRV,PPNAM)                                     2NOV    108
      CALL DMSET(ICOM1+4,ICRV,PPVAL)                                     2NOV    109
      CALL DMSET(ICOM1+5,ICRV,NPTS)                                      TEMP80    8
      CALL DMSET(ICOM1+6,ICRV,ISYM)                                      TEMP80    9
      CALL DMSET(ICOM1+10,ICRV,IV)                                       TEMP80   10
      CALL DMGT80(NPTS,IREC)                                             TEMP80   11
      DO 10 I=1,NPTS                                                     TEMP80   12
   10 NREC(I)=1                                                          TEMP80   13
      CALL DMSET(ICOM1+11,ITEM80,ICRV)                                   TEMP80   14
      JJ=NPTS*2**18                                                      TEMP80   15
      CALL DMSET(ICOM7+JJ,IREC,NREC)                                     TEMP80   16
      CALL DMGT80(NPTS,IX)                                               TEMP80   17
      CALL DMGT80(NPTS,IY)                                               TEMP80   18
      CALL DMSET(ICOM7+JJ,IX,XIN)                                        TEMP80   19
      CALL DMSET(ICOM7+JJ,IY,YIN)                                        TEMP80   20
      CALL DMSET(ICOM1+7,ICRV,IREC)                                      TEMP80   21
      CALL DMSET(ICOM1+8,ICRV,IX)                                        TEMP80   22
      CALL DMSET(ICOM1+9,ICRV,IY)                                        TEMP80   23
      CALL DMGT80(7,IAXIS)                                               TEMP80   24
      CALL DMSET(ICOM1+7,ITEM80,IAXIS)                                   TEMP80   25
      CALL DMSET(ICOM1+5,IAXIS,XNAME)                                    TEMP80   26
      CALL DMSET(ICOM1+7,IAXIS,YNAME)                                    TEMP80   27
      CALL DMGT80(8,IAXISC)                                              TEMP80   28
      CALL DMSET(ICOM1+9,ITEM80,IAXISC)                                  TEMP80   29
      JJ=8*2**18                                                         TEMP80   30
      CALL DMSET(ICOM7+JJ,IAXISC,ASCL)                                   TEMP80   31
      END                                                                TEMP80   32
```

```
1                SUBROUTINE RLTEMP(IT8D)                                          RLTEMP     2
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG         30AUG      1
           C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)   /DISKT/    3
                 CALL DMGET(ICOM1+7,IT8D,IAX)                                     RLTEMP     4
5                CALL DMGET(ICOM1+9,IT8D,IAXSC)                                   RLTEMP     5
                 CALL DMGET(ICOM1+11,IT8D,ICRV)                                   RLTEMP     6
                 CALL DMRL8D(IAX,IAXSC,ICRV,IT8D)                                 RLTEMP     7
                 END                                                              RLTEMP     8
```

```
      SUBROUTINE SAVET(IPLOT,NPTS)                                          SAVET     2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG              30AUG     1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)        /CISKT/   3
      COMMON /TTEMP/TEMP(10)                                                /TTEMP/   2
      COMMON/TTABLE/CL(20),DELC(20),ALPHA(20),CD(20)                        /TTABLE/  2
      CALL DMGTBD(10,IBEAD)                                                 SAVET     5
      CALL DMSET(ICOM1+10,IBEAD,IBEAD)                                      SAVET     6
      CALL DMGTBD(NPTS,TEMP(1))                                             SAVET     7
      CALL DMGTBD(NPTS,TEMP(2))                                             SAVET     8
      CALL DMGTBD(NPTS,TEMP(3))                                             SAVET     9
      CALL DMGTBD(NPTS,TEMP(4))                                             SAVET    10
      TEMP(5)=NPTS                                                          SAVET    11
      CALL DMSET(ICOM1+14,IPLOT,TEMP(6))                                    SAVET    12
      CALL DMSET(ICOM1+13,IPLOT,TEMP(7))                                    SAVET    13
      J=7*2**18                                                             SAVET    14
      CALL DMSET(ICOM7+J,IBEAD,TEMP)                                        SAVET    15
      CALL DMSET(ICOM1+5,IBEAD,NPTS)                                        22JUNE   27
      J=NPTS*2**18                                                          SAVET    16
      CALL DMSET(ICOM7+J,TEMP(1),CL)                                        SAVET    17
      CALL DMSET(ICOM7+J,TEMP(2),DELC)                                      SAVET    18
      CALL DMSET(ICOM7+J,TEMP(3),ALPHA)                                     SAVET    19
      CALL DMSET(ICOM7+J,TEMP(4),CD)                                        SAVET    20
      IF(IBD1.EQ.0)GO TO 50                                                 SAVET    21
      CALL LINK(IBD1,IBEAD,8,9)                                             SAVET    22
      RETURN                                                                SAVET    23
   50 IBD1=IBEAD                                                            SAVET    24
      END                                                                   SAVET    25
```

442

```
      SUBROUTINE PCURVE(IBEAD1,IBEAD,IVALU)                              PCURVE      2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG       1
C     J=N*2**1?  CMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/     3
      COMMON/TDIS/ITRIM1,FRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,       T30AUG      21
     1XDATMAX,YDATMIN,YDATMAX                                            T30AUG      22
      COMMON /TTEMP/TEMP(10)                                             /TTEMP/     2
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/    2
      DATA TEST/8HINACTIVE/,IX/1HX/                                      22JUNE      29
      TEE=0                                                              PCURVE      7
      CALL CMGET(ICOM1+12,IBEAD,ACT)                                     PCURVE      8
      IF(ACT.EQ.TEST)RETURN                                              22JUNE      30
      CALL CMGET(ICOM1+3,IBEAD,PRNAME)                                   22JUNE      31
      CALL CMGET(ICOM1+5,IBEAD,NPOINTS)                                  PCURVE      10
      CALL FITBLK(IBEAD,NP,NDEG,IFIT,IDVAR)                              16AUG76     3
      YVAL=YVA(NP,NDEG,IFIT,6,IDVAR,IER,XVAL)                            10MAY77     54
   20 CONTINUE                                                           14JULY76    2
      IC=IC+1                                                            PCURVE      18
      XX(IC)=YVAL                                                        PCURVE      19
      CALL CMGET(ICOM1+4,IBEAD,YY(IC))                                   PCURVE      20
      END                                                                PCURVE      21
```

```
      SUBROUTINE ASCALE(XA,YA,NPA)                                      ASCALE     2
      COMMON/TOIS/ITRIM1,PRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,      T3DAUG    21
     1XDATMAX,YDATMIN,YDATMAX                                           T3DAUG    22
      DIMENSION XA(1),YA(1)                                             ASCALE     4
      DATA XL/9./,YL/6./                                                ASCALE     5
      XMIN=AMIN(XA,NPA)                                                 ASCALE     6
      XMAX=AMAX(XA,NPA)                                                 ASCALE     7
      YMIN=AMIN(YA,NPA)                                                 ASCALE     8
      YMAX=AMAX(YA,NPA)                                                 ASCALE     9
      XDATMIN=XMIN                                                      T3DAUG     8
      XDATMAX=XMAX                                                      T3DAUG     9
      YDATMIN=YMIN                                                      T3DAUG    10
      YDATMAX=YMAX                                                      T3DAUG    11
      DDX=.1*(XMAX-XMIN)                                                ASCALE    10
      DDY=.1*(YMAX-YMIN)                                                ASCALE    11
      XMIN=XMIN-DDX                                                     ASCALE    12
      YMIN=YMIN-DDY                                                     ASCALE    13
      XMAX=XMAX+DDX                                                     ASCALE    14
      YMAX=YMAX+DDY                                                     ASCALE    15
      DELX=(XMAX-XMIN)/XL                                               ASCALE    16
      DELY=(YMAX-YMIN)/YL                                               ASCALE    17
      END                                                               ASCALE    18
```

444

```
1                FUNCTION AMAX(A,N)                                    AMAX      2
                 DIMENSION A(N)                                        AMAX      3
                 V=A(1)                                                AMAX      4
                 DO 10 I=2,N                                           AMAX      5
5                IF(A(I).GT.V)V=A(I)                                   AMAX      6
            10   CONTINUE                                              AMAX      7
                 AMAX=V                                                AMAX      8
                 END                                                   AMAX      9
```

```
1                   SUBROUTINE LINK(IBD1,IBONEW,JPREV,JNEXT)                    LINK        2
                C   ATTACHES LINK IBONEW TO STRING BEGINNING WITH IBD1          LINK        3
                C   POINTER TO PREVIOUS BEAD IN WORD JPREV                      LINK        4
                C   POINTER TO NEXT BEAD IN WORD JNEXT                          LINK        5
5                   DATA ICOMP/3100000000003/                                   LINK        6
                    NEXT=IBD1                                                   LINK        7
                    IPREV=IBD1                                                  LINK        8
                    IF(IBD1*IBONEW.NE. 0)GO TO 10                               LINK        9
                    WRITE (6,5)                                                 CYBTTT      24
10              5   FORMAT(28H NULL BEAD ADDRESSES )                            LINK       11
                    STOP                                                        LINK       12
                10  CONTINUE                                                    LINK       13
                    CALL CMGET(ICOMP+JNEXT,NEXT,NEXTAD)                         LINK       14
                    IF(NEXTAD.EQ.0)GO TO 20                                     LINK       15
15                  IPREV=NEXTAD                                                LINK       16
                    NEXT=NEXTAD                                                 LINK       17
                    GO TO 10                                                    LINK       18
                20  CALL CMSET(ICOMP+JNEXT,NEXT,IBONEW)                         LINK       19
                    CALL CMSET(ICOMP+JPREV,IBONEW,IPREV)                        LINK       20
20                  END                                                         LINK       21
```

```
      SUBROUTINE SORT2(X,Y,NPOINTS,IV)                                  SORT2    2
C     MODIFICATION OF SORTER                                            SORT2    3
C                                                                       SORT2    4
C         THIS ROUTINE SORTS THE COORDINATE ARRAYS                      SORT2    5
C         THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT        SORT2    6
C         VARIABLE.                                                     SORT2    7
C                                                                       SORT2    8
C         NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.  SORT2    9
C              IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE SORT2   10
C                       TWO ARRAYS ARE SORTED SO THAT THE             SORT2   11
C                       X - COORDINATE ARRAY IS IN ASCENDING ORDER.   SORT2   12
C                       IF IV = 1HY, THEN THE TWO ARRAYS ARE SORTED SO SORT2   13
C                       THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER. SORT2 14
C                                                                       SORT2   15
      DIMENSION X(1),Y(1)                                               SORT2   16
      DO 10 L=1,NPOINTS,L                                               SORT2   17
      M = 2 * L - 1                                                     SORT2   18
   10 CONTINUE                                                          SORT2   19
C                                                                       SORT2   20
C         START LOOP                                                    SORT2   21
C                                                                       SORT2   22
   20 M = M / 2                                                         SORT2   23
      IF(M .EQ. 0) GO TO 70                                             SORT2   24
      K = NPOINTS - M                                                   SORT2   25
      DO 60 J=1,K                                                       SORT2   26
      L = J                                                             SORT2   27
   30 IF(L .LT. 1) GO TO 60                                             SORT2   28
C                                                                       SORT2   29
C         USE THE SORT KEY.                                             SORT2   30
C                                                                       SORT2   31
      IF(IV .EQ. 1HY) GO TO 40                                          SORT2   32
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORT2   33
      GO TO 50                                                          SORT2   34
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORT2   35
C                                                                       SORT2   36
C         EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.               SORT2   37
C                                                                       SORT2   38
   50 TEMP = X(L)                                                       SORT2   39
      X(L) = X(L+M)                                                     SORT2   40
      X(L+M) = TEMP                                                     SORT2   41
C                                                                       SORT2   42
C         EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.               SORT2   43
C                                                                       SORT2   44
      TEMP = Y(L)                                                       SORT2   45
      Y(L) = Y(L+M)                                                     SORT2   46
      Y(L+M) = TEMP                                                     SORT2   47
C                                                                       SORT2   48
      L = L - M                                                         SORT2   49
      GO TO 30                                                          SORT2   50
   60 CONTINUE                                                          SORT2   51
      GO TO 20                                                          SORT2   52
   70 RETURN                                                            SORT2   53
      END                                                               SORT2   54
```

```
      SUBROUTINE SORTCK2(ICRVPTR,IVK,NPOINTS)                                  SORTCK2    2
      COMMON/DISKT/I801,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG                 30 AUG     1
C     J=N*2**18  DMSET(ICOM7+J,I80A,DATA)  STORES DATA(N) IN I80A(1)           /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                    /POINTS/   2
C     MODIFICATION OF SORTCK                                                   SORTCK2    5
C                                                                              SORTCK2    6
C         THIS ROUTINE MOVES THE COORDINATE ARRAYS AND THE STATUS ARRAY        SORTCK2    7
C         FROM THE DATA MANAGER FILE INTO CORE AND CHECKS TO SEE IF THE        SORTCK2    8
C         POINTS HAVE TO BE SORTED. IF SO, THEN THEY ARE SORTED                SORTCK2    9
C                                                                              SORTCK2   10
C         ICRVPTR - I - THE POINTER TO THE CURVE HEAD OF THE CURVE WHICH       SORTCK2   11
C                   IS TO BE EXAMINED FOR SORT ORDER.                          SORTCK2   12
C             IVK - I - THE INDEPENDENT VARIABLE SUBMITTED BY THE USER.        SORTCK2   13
C         NPOINTS - O - THE NUMBER OF POINTS IN THE COORDINATE BEADS.          SORTCK2   14
C                                                                              SORTCK2   15
C                                                                              SORTCK2   16
C         MOVE THE COORDINATE ARRAYS AND THE STATUS ARRAY INTO CORE.           SORTCK2   17
C                                                                              SORTCK2   18
      CALL DMGET(ICOM1+5,ICRVPTR,NPOINTS)                                      SORTCK2   19
      CALL DMGET(ICOM1+7,ICRVPTR,IRECPTR)                                      SORTCK2   20
      CALL DMGET(ICOM1+8,ICRVPTR,IXPTR)                                        SORTCK2   21
      CALL DMGET(ICOM1+9,ICRVPTR,IYPTR)                                        SORTCK2   22
      KOMPCOD = 700000000018 + SHIFT(NPOINTS,18)                               SORTCK2   23
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                         SORTCK2   24
      CALL DMGET(KOMPCOD,IXPTR,X)                                              SORTCK2   25
      CALL DMGET(KOMPCOD,IYPTR,Y)                                              SORTCK2   26
C                                                                              SORTCK2   27
C         CHECK TO SEE IF THE POINTS NEED TO BE SORTED. THEY SHOULD BE         SORTCK2   28
C         SORTED WITH THE SORT KEY EQUAL TO THE INDEPENDENT VARIABLE           SORTCK2   29
C         SUBMITTED BY THE USER. IF THEY ARE ALREADY SORTED THAT WAY, NO       SORTCK2   30
C         SORT IS NECESSARY. IF NOT, THEY MUST BE SORTED.                      SORTCK2   31
C                                                                              SORTCK2   32
      CALL DMGET(ICOM1+10,ICRVPTR,IVAR)                                        SORTCK2   33
      IF(IVAR .NE. 0) GO TO 10                                                 SORTCK2   34
      IVAR = 1HX                                                               SORTCK2   35
      CALL DMSET(ICOM1+10,ICRVPTR,IVAR)                                        SORTCK2   36
   10 IF(IVK .EQ. IVAR) RETURN                                                 SORTCK2   37
C                                                                              SORTCK2   38
C         THE POINTS ARE NOT SORTED AS SPECIFIED BY THE USER. SORT             SORTCK2   39
C         THEM.                                                                SORTCK2   40
C                                                                              SORTCK2   41
      CALL SORT2(X,Y,NPOINTS,IVK)                                              SORTCK2   42
      RETURN                                                                   SORTCK2   43
      END                                                                      SORTCK2   44
```

```
      SUBROUTINE SPLICE(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)                SPLICE     2
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU          SPLICE     3
C     IN WORD JMATCH OF EACH BEAD.                                            SPLICE     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND                   SPLICE     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                          SPLICE     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                      SPLICE     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                                  SPLICE     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                      SPLICE     9
      DATA ICOMP/O1000000000008/                                              SPLICE    10
      NEXT=IBD1                                                               SPLICE    11
      IPREV=IBD1                                                              SPLICE    12
      IF(IBD1.NE.0)GOTO10                                                     SPLICE    13
      PRINT                                                                   SPLICE    14
    5 FORMAT(13H IBD1 IS NULL )                                               SPLICE    15
      IS=2                                                                    SPLICE    16
      RETURN                                                                  SPLICE    17
   10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                                   SPLICE    18
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                     SPLICE    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                             SPLICE    20
      IF(NEXTAD.EQ.0)GOTO12                                                   SPLICE    21
      IPREV=NEXT                                                              SPLICE    22
      NEXT=NEXTAD                                                             SPLICE    23
      GO TO 10                                                                SPLICE    24
   12 WRITE (6,6) IVALU                                                       CY8777    21
    6 FORMAT(19H NO MATCH FOR IVALU ,O20)                                     SPLICE    26
      IS=2                                                                    SPLICE    27
      RETURN                                                                  SPLICE    28
   15 CONTINUE                                                                SPLICE    29
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL               SPLICE    30
      IF(NEXT.EQ.IBD1)GO TO 25                                                SPLICE    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE           SPLICE    32
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                         SPLICE    33
      IF(NEXTAD.EQ.0)GOTO22                                                   SPLICE    34
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                     SPLICE    35
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                        SPLICE    36
   22 CONTINUE                                                                SPLICE    37
      CALL SUBR(IBD1,NEXT,IVALU)                                              SPLICE    38
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                            SPLICE    39
      RETURN                                                                  SPLICE    40
   25 CALL SUBR(IBD1,IBD1,IVALU)                                              SPLICE    41
      IF(NEXTAD.EQ.0)GO TO 40                                                 SPLICE    42
      NEXT=NEXTAD                                                             SPLICE    43
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                     SPLICE    44
      CALL SUBR(IBD1,NEXT,IVALU)                                              SPLICE    45
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                            SPLICE    46
      IF(NEXTAD.EQ.0)GO TO 40                                                 SPLICE    47
      NEXT=NEXTAD                                                             SPLICE    48
      GO TO 30                                                                SPLICE    49
   40 CONTINUE                                                                SPLICE    50
      IF(IS.EQ.1)CALL DMRLBD(IBD1)                                            SPLICE    51
      END                                                                     SPLICE    52
```

```
      SUBROUTINE SQUEEZ(NPOINTS,NVALID)                                  SQUEEZ     2
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/   2
      J=0                                                                SQUEEZ     4
      DO 10 I=1,NPOINTS                                                  SQUEEZ     5
      IF(NREC(I).LE.0)GO TO 10                                           SQUEEZ     6
      J=J+1                                                              SQUEEZ     7
      X(J)=X(I)                                                          SQUEEZ     8
      Y(J)=Y(I)                                                          SQUEEZ     9
   10 CONTINUE                                                           SQUEEZ    10
      NVALID=J                                                           SQUEEZ    11
      END                                                                SQUEEZ    12
```

```
1                SUBROUTINE TABRL(IBEAD1,IBEAD,IVALU)                                TABREL     2
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG            30AUG      1
               C J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)      /DISKT/    3
                 DO 5 J=1,4                                                          TABREL     4
5                CALL DMGET(ICOM1+J,IBEAD,IBDAR)                                     TABREL     5
                 CALL DMRLBD(IBDAR)                                                  TABREL     6
               5 CONTINUE                                                            TABREL     7
                 END                                                                 TABREL     8
```

```
      SUBROUTINE FITBLK (ICVBD,N,NDEG1,IFIT,IV1)                        FIX2      4
C     IFIT, 1 LSQ, 2 SPLINE, 3 AITKEN                                   16AUG76   17
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/  2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG     1
C     J=N**2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)   /DISKT/   3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1    10MAY77   61
     1,IDELTF,IDELBC(2)                                                 FLCAD2    5
      IFIT=0                                                            16AUG76   20
      CALL DMGET(ICOM1+11,ICVBD,IFITBD)                                 16AUG76   21
      IF(IFITBD.NE.0)GO TO 2                                            5OCT      101
      CALL ERRDFM(-3.,3.,34HCURVE FITS REQUIRED FOR CROSSPLOTS ,34,     5OCT      102
     1 6HFITBLK,6HINFORM)                                               5OCT      103
      CALL RETSKC(7HQTRBACK)                                            5OCT      104
    2 CONTINUE                                                          5OCT      105
      CALL DMGET(ICOM1+1,IFITBD,ITYPE)                                  16AUG76   23
      IF(ITYPE.EQ.1HLEASTSQUAR)IFIT=1                                   16AUG76   24
      IF(ITYPE.EQ.6HSPLINE)IFIT=2                                       16AUG76   25
      IF(ITYPE.EQ.6HAITKEN)IFIT=3                                       16AUG76   26
      CALL DMGET(ICOM1+5,IFITBD,N)                                      16AUG76   27
      JJ=N*2**18                                                        16AUG76   28
      CALL DMGET (ICOM1+10,ICVBD,IV1)                                   FIX2      5
      GO TO (10,20,30)IFIT                                              16AUG76   30
   10 CONTINUE                                                          16AUG76   31
      CALL DMGET(ICOM7+JJ+5,IFITBD,NREC)                                16AUG76   32
      IF (JT.NE.3.AND.JT.NE.5) GO TO 11                                 LASTBUX   11
      CALL DMGET (ICOM1+5,ICVBD,N)                                      LASTBUX   12
      JJ=N*2**19                                                        LASTBUX   13
C                                                                       LASTBUX   14
      CALL DMGET (ICOM1+8,ICVBD,IXPTR)                                  LASTBUX   15
      CALL DMGET (ICOM1+9,ICVBD,IYPTR)                                  LASTBUX   16
      CALL DMGET (ICOM7+JJ,IXPTR,X)                                     LASTBUX   17
      CALL DMGET (ICOM7+JJ,IYPTR,Y)                                     LASTBUX   18
   11 CONTINUE                                                          LASTBUX   19
      RETURN                                                            16AUG76   33
   20 CONTINUE                                                          16AUG76   34
      CALL DMGET(ICOM7+JJ+5,IFITBD,NREC)                                16AUG76   35
      KK=JJ+N+5                                                         16AUG76   36
      LL=JJ+2*N+5                                                       16AUG76   37
      CALL DMGET(ICOM7+KK,IFITBD,X)                                     16AUG76   39
      CALL DMGET(ICOM7+LL,IFITBD,Y)                                     16AUG76   40
      RETURN                                                            16AUG76   41
   30 CALL DMGET (ICOM1+6,IFITBD,NDEG1)                                 FIX2      6
      KK=JJ+6                                                           16AUG76   46
      LL=JJ+N+6                                                         16AUG76   47
      CALL DMGET(ICOM7+KK,IFITBD,X)                                     16AUG76   49
      CALL DMGET(ICOM7+LL,IFITBD,Y)                                     16AUG76   50
      RETURN                                                            16AUG76   51
      END                                                               16AUG76   54
```

```
      FUNCTION YVA(NP,NOEG,IFIT,ICODE,IV,IER,XX)                        16AUG76   55
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
C     IFIT, 1 LEAST SQUARES, 2 SPLINE, 3 AITKEN                         26AUG      18
C     ICODE, 1 COORD, 2 SLOPES, 3 MIN, 4 DELTA, 5 MAX, 6 TRIM           10MAY77    62
C     IER GT 1 MEANS BAD INTERP                                         16AUG76    59
C     (IER.EQ.6)MEANS ARGUMENT OUT OF RANGE                             29APR      23
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        29APR      24
C                                                                       19APR      1
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        19APR      2
C                                                                       19APR      3
C     IER=0  SUCCESSFUL INTERPOLATION                                   19APR      4
C     IER=1  SUCCESSFUL EXTRAPOLATION                                   19APR      5
C     IER=2  ARGUMENT OUT OF BOUNDS                                     19APR      6
C     IER=3  ATTEMPTING CURVE EVALUATION WITHOUT FIT                    19APR      7
C     IER=4  ATTEMPTING SLOPES CROSSPLOT WITHOUT SPLINE FIT             19APR      8
      IF (IFIT.NE.2) GO TO 120                                          29APR      25
      IF(IV.EQ.1HY)GO TO 105                                            29APR      26
      IF(XX.LT.X(1))GO TO 110                                           29APR      27
      IF(XX.GT.X(NP))GO TO 110                                          29APR      28
      GO TO 120                                                         29APR      29
  105 IF(XX.LT.Y(1))GO TO 110                                           29APR      30
      IF(XX.GT.Y(NP))GO TO 110                                          29APR      31
      GO TO 120                                                         29APR      32
  110 IER=6                                                             29APR      33
      RETURN                                                            29APR      34
  120 CONTINUE                                                          29APR      35
      XV=-4.                                                            19APR
      YV=0.                                                             19APR
      IF(IFIT.LT.1)GO TO 60                                             16AUG76    62
      IF(IFIT.GT.3)GO TO 60                                             16AUG76    63
      IER=0                                                             16AUG76    64
      GO TO (2,12,30,2,28,2)ICODE                                       10MAY77    63
    2 CONTINUE                                                          16AUG76    66
      GO TO (10,15,35)IFIT                                              16AUG76    67
   10 YY=P(XX,NREC,NP)                                                  16AUG76    68
      GO TO 50                                                          16AUG76    69
   12 NTYP=2                                                            16AUG76    70
      IF(IFIT.NE.1)GO TO 20                                             19APR      11
      YY=PP(XX,NREC,NP)                                                 19APR      12
      GO TO 50                                                          19APR      13
   15 NTYP=1                                                            16AUG76    72
      GO TO 20                                                          16AUG76    73
   20 CONTINUE                                                          26AUG      19
      IF(IFIT.EQ.2)GO TO 21                                             19APR      14
C     SLOPES CROSSPLOT PERMITTED ONLY WITH SPLINE CURVEFIT              19APR      15
      CALL ERRORM(XV,YV-1.,41HSLOPES CROSSPLOT REQUIRES SPLINE CURVEFIT, 19APR      16
     141,3HYVA    ,6HINFORM)                                            19APR      17
      CALL ABTSKC(7HQTRBACK)                                            14JUNE     72
   21 CONTINUE                                                          19APR      20
   50 IF(IV.EQ.1HY)GO TO 25                                             26AUG      20
      YY=SPLIN2(XX,NP,X,Y,NREC,NTYP)                                    26AUG      21
      GO TO 50                                                          26AUG      22
   25 YY=SPLIN2(XX,NP,Y,X,NREC,NTYP)                                    26AUG      23
      GO TO 50                                                          16AUG76    75
   28 CONTINUE                                                          10MAY77    64
      IF(IV.NE.1HY)GO TO 29                                             10MAY77    65
      YY=AMAX(X,NP)                                                     10MAY77    66
```

```
                      GO TO 50                                                  10MAY77    67
                29    YY=AMAX(Y,NP)                                             10MAY77    68
60                    GO TO 50                                                  10MAY77    69
                30    CONTINUE                                                  26AUG      24
                      IF(IV.NE.1HY)GO TO 32                                     20SEPT     152
                      YY=AMIN(X,NP)                                             26AUG      26
                      GO TO 50                                                  26AUG      27
65              32    YY=AMIN(Y,NP)                                             26AUG      28
                      GO TO 50                                                  16AUG76    77
                35    CONTINUE                                                  26AUG      29
                      IF(IV.EQ.1HY)GO TO 37                                     26AUG      30
                      CALL INTERP (X,Y,NP,NDEG,XX,YY,NREC,IER)                  LASTBUX    31
70                    GO TO 50                                                  26AUG      32
                37    CALL INTERP (Y,X,NP,NDEG,XX,YY,NREC,IER)                  LASTBUX    33
                      GO TO 50                                                  16AUG76    79
                50    YVA=YY                                                    16AUG76    80
                      IF(IER.EQ.0)RETURN                                        19APR      21
75                    IF(IER.EQ.1)GO TO 49                                      19APR      22
                      CALL ERROFM(XV,YV-1.,21HARGUMENT OUT OF RANGE  ,21, 3HYVA  ,  19APR      23
                     15HINFORM)                                                 19APR      24
                      CALL AETSKC(7HQTRBACK)                                    14JUNE     73
                49    CONTINUE                                                  19APR      26
80                    RETURN                                                    16AUG76    81
                60    IER=3                                                     19APR      29
                      CALL ERROFM(XV,YV-1.,40HATTEMPTING TO EVALUATE CURVE WITHOUT FIT ,  19APR      30
                     140,3HYVA  ,6HINFORM)                                      19APR      31
                      CALL AETSKC(7HQTRBACK)                                    14JUNE     74
85                    END                                                       16AUG76    83
```

```
                      FUNCTION PP(Z,C,N)                                        19APR  32
C                     COMPUTES DERIVATIVE OF POLYNOMIAL IN Z WITH COEFFICIENTS IN C  19APR  33
C                     DEGREE IS N-1                                             19APR  34
                      DIMENSION C(N)                                            19APR  35
                      PP=C(2)                                                   19APR  36
                      DO 10 I=3,N                                               19APR  37
                      PP=PP+C(I)*Z**(I-2)                                       19APR  38
                   10 CONTINUE                                                  19APR  39
                      RETURN                                                    19APR  40
                      END                                                       19APR  41
```

```
1                FUNCTION AMIN(A,N)                                  AMIN        2
                 DIMENSION A(N)                                      AMIN        3
                 V=A(1)                                              AMIN        4
                 DO 10 I=2,N                                         AMIN        5
5                IF(A(I).LT.V)V=A(I)                                 AMIN        6
              10 CONTINUE                                            AMIN        7
                 AMIN=V                                              AMIN        8
                 END                                                 AMIN        9
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C  - -  A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE     SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 8 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 8                                          SPLIN2    27
       KK = K                                                            SPLIN2    28
       GO TO 120                                                         SPLIN2    29
    8 CONTINUE                                                           SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN                                                             SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1        1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3   SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
      RETURN                                                             SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
      FUNCTION P(Z,C,N)                                                 P          2
C                                                                       P          3
C         THIS ROUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES P          4
C         POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".          P          5
C                                                                       P          6
C         Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                P          7
C         C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.   P          8
C         N - I - THE NUMBER OF COEFFICIENTS.                           P          9
C                                                                       P         10
      DIMENSION C(N)                                                    P         11
      NDEG = N - 1                                                      P         12
      P=C(1)                                                        18JUNE76      21
      DO 10 I=2,N                                                   18JUNE76      22
      P=P+C(I)*Z**(I-1)                                             18JUNE76      23
   10 CONTINUE                                                          P         16
      RETURN                                                            P         17
      END                                                               P         18
```

```
      SUBROUTINE RWPLGR(IBEAD1,INEXT,IPREV,IONE)                        RWPLGR     2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBD4,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      EXTERNAL SUBGET                                                   RWPLGR     5
C     IONE=1 MEANS ONLY ONE PLOT IN GROUP                               RWPLGR     6
      IONE=0                                                            RWPLGR     7
      CALL DMGET(ICOM1+IPREV,IBEAD1,IPR)                                RWPLGR     8
      IF(IPR.NE.0)GO TO 10                                              RWPLGR     9
   10 CALL DMGET(ICOM1+INEXT,IBEAD1,NEXT)                               RWPLGR    10
      IF(NEXT.NE.0)RETURN                                               RWPLGR    11
      IONE=1                                                            RWPLGR    12
      RETURN                                                            RWPLGR    13
   10 CALL SPLICE(IBEAD1,SUBGET,IPREV,0,IPREV,INEXT,0)                  RWPLGR    14
      END                                                               RWPLGR    15
```

```
1                SUBROUTINE SUBGET(IBEAD1,IBEAD,IVALU)                    RWFLGR     16
                 IBEAD1=IBEAD                                             RWFLGR     17
                 END                                                      RWFLGR     18
```

```
      SUBROUTINE VARIN(ITITLE,VALUE,N,IXDGU,IYDGU,IDDAD)                VARIN     2
C                                                                       VARIN     3
C     ITITLE = ARRAY OF VARIABLE NAMES (10 CHAR WORDS - ONE NAME/WORD)  VARIN     4
C     VALUE  = ARRAY OF CURRENT PARAMETER VALUES                        VARIN     5
C     N      = NUMBER OF ITEMS TO BE DISPLAYED                          VARIN     6
C     IXDGU,IYDGU  =  UPPER LEFT CORNER OF TABLE (DGUS)                 VARIN     7
C     IDDAD  = ARRAY TO TEMP. STORE ASSOCIATED ADDRESSES (DIMENSION N+1) VARIN     8
C                                                                       VARIN     9
C     NAMED COMMON FOR GENERAL GRAPHICS CONSTANTS                       VARIN    10
C                                                                       VARIN    11
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777    1
     Y    ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77   3
C                                                                       VARIN    13
C                                                                       VARIN    15
      DIMENSION ITITLE(1),VALUE(1),IDDAD(1)                             VARIN    16
C                                                                       VARIN    17
      DATA KSAVE/1/                                                     26AUG    17
      KD=1                                                              VARIN    19
      NBYTE=0                                                           VARIN    20
      IX1=IXDGU                                                         VARIN    21
      IX2=IX1+312                                                       VARIN    22
      IX3=IX2+312                                                       VARIN    23
      DO 1 I=1,N                                                        VARIN    24
      ENCODE(10,100,IDUM) VALUE(I)                                      VARIN    25
  100 FORMAT(G10.3)                                                     VARIN    26
      IY1=IYDGU+75-75*I                                                 VARIN    27
      CALL GURSET(IX1,IY1,0038,IBUF,NBYTE,MBYTE)                        VARIN    28
      CALL GUAN(ITITLE(I),10,IBUF,NBYTE,MBYTE)                          VARIN    29
      CALL GURSET(IX2,IY1,1038,IBUF,NBYTE,MBYTE)                        VARIN    30
      CALL GUAN(IDUM,10,IBUF,NBYTE,MBYTE)                               VARIN    31
      KD=KD+1                                                           VARIN    32
    1 CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(KD),IBUT,KD,-0)                 VARIN    33
C                                                                       VARIN    34
      CALL GURSET(IXDGU+216,IYDGU+75-75*(N+1),103B,IBUF,NBYTE,MBYTE)    VARIN    35
      CALL GUAN(6HACCEPT,6,IBUF,NBYTE,MBYTE)                            VARIN    36
      KD=KD+1                                                           VARIN    37
      KEND=KD                                                           VARIN    38
      CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(KD),IBUT,KD,-0)                 VARIN    39
C                                                                       VARIN    40
    2 CALL GIBUT(0,NCON,IDOT,IDDC,-0)                                   VARIN    41
      IF(IDDC .GE. KSAVE+1 .AND. IDDC .LE. KEND-1) GO TO 5              VARIN    42
      IF(IDDC .EQ. KEND) GO TO 3                                        VARIN    43
      GO TO 2                                                           VARIN    44
    3 NE=KEND-KSAVE                                                     VARIN    45
      DO 4 I=1,NE                                                       VARIN    46
    4 CALL GIERAS(IDDAD(KSAVE+I),-0)                                    VARIN    47
      KD=KSAVE                                                          VARIN    48
      RETURN                                                            VARIN    49
C                                                                       VARIN    50
    5 ICHANG=IDDC                                                       VARIN    51
      IY2=IYDGU+75-75*(IDDC-KSAVE)                                      VARIN    52
    6 NL=10                                                             VARIN    53
      CALL GIEOM (NCON,IEOM,IBUT,KEND+1,-0)                             CY8777   25
      CALL GIANS(NCON,NL,IX3,IY2)                                       VARIN    55
C                                                                       VARIN    56
    7 CALL GIBUT(0,NCON,IDOT,IDDC,-0)                                   VARIN    57
      IF(IDDC .EQ. KEND+1) GO TO 9                                      VARIN    58
```

```
                GO TO 7                                                  VARIN     59
              C                                                          VARIN     60
60          8 CALL GIANE(NCON,NL,LREG)                                   VARIN     61
                IF(NL .LT. 1) GO TO 6                                    VARIN     63
                ENCODE (7,12A,IFMT) NL                                   VARIN     64
             12A FORMAT(2H(F,I2,3H.0))                                   VARIN     65
                II=ICHANG-KSAVE                                          VARIN     66
65              DECODE(NL,IFMT,LREG) VALUE(II)                           VARIN     67
                CALL ESCK,RETURNS(6)                                     SOCT     107
              C                                                          VARIN     71
             20 ENCODE(10,100,IDUM) VALUE(II)                            VARIN     72
                CALL GIEPAS(IDDAD(ICHANG),-0)                            VARIN     73
70            C                                                          VARIN     74
                CALL CURSET(IX1,IY2,0033,IBUF,NBYTE,MBYTE)               VARIN     75
                CALL GUAN(ITITLE(II),10,IBUF,NBYTE,MBYTE)                VARIN     76
                CALL CURSET(IX2,IY2,1033,IBUF,NBYTE,MBYTE)               VARIN     77
                CALL GUAN(IDUM,10,IBUF,NBYTE,MBYTE)                      VARIN     78
75              CALL GIDISP(NCON,IBUF,NBYTE,IDDAD(ICHANG),IBUT,ICHANG,-0) VARIN    79
              C                                                          VARIN     80
                GO TO 2                                                  VARIN     81
              C                                                          VARIN     82
                END                                                      VARIN     83
```

```
      OVERLAY(12,0)                                                          DTRIM      2
      PROGRAM DTRIM                                                          DTRIM      3
      COMMON/DISKT/I8D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG      1
C     J=N*2**13  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)         /DISKT/    3
      COMMON/OPTION/IOPTION,ISHINE                                           LASTBUX    1
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR      /BEADS/    2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                     /ITRIMT/   2
      EXTERNAL TDISP                                                         DTRIM      8
      EXTERNAL RLSPM,PRNTPLT                                                 DTRIM      9
      COMMON/TDIS/ITRIM1,PRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,           T30AUG     21
     1XDATMAX,YDATMIN,YDATMAX                                                T30AUG     22
C     PICK X,PICK Y                                                          DTRIM      11
      CALL ORIGIN(-4.,2.)                                                    1SEPT      18
      CALL PTPLOT(NX,NY)                                                     1SEPT      19
C     DISPLAY PLOT INFO,CAN CHANGE AS DESIRED                                DTRIM      12
C     SET OPTION=CROSSPLOT                                                   DTRIM      13
C     CHOOSE X,Y,SCALING HERE                                                DTRIM      14
      IOPTION =10HCROSS PLOT                                                 DTRIM      15
C     RESTORE IPLTPTR                                                        DTRIM      16
      IF(IPLTPTR.EQ.ITRIM1)IPLTPTR=IPLTSAV                                   DTRIM      17
      IF(ITRIM1.EQ.0)GO TO 20                                                DTRIM      18
C     IF PRIOR TRIM STORED,RELEASE IT                                        DTRIM      19
      CALL DMSET (ICOM1+10,ITRIM1,IPLT)                                      25AUG      36
      IF (IPLT.NE.99) GO TO 19                                               25AUG      37
      GO TO 20                                                               25AUG      38
   19 CALL ITRIMR (ITRIM1)                                                   25AUG      39
      CALL RLSPLT(ITRIM1)                                                    5OCT       56
   20 CONTINUE                                                               DTRIM      22
      CALL SPLICE(I8D1,TDISP,10,I3D1,9,8,0)                                  DTRIM      23
C     PRINT TRIM PLOTS                                                       DTRIM      24
C     DISPLAYS TRIM PLOTS                                                    DTRIM      27
C                                                                            DTRIM      28
C     SAVE IPLTPTR                                                           DTRIM      29
      IPLTSAV=IPLTPTR                                                        DTRIM      30
      IPLTPTR=ITRIM1                                                         DTRIM      31
      CALL AETSKC (7HFRSTPL2)                                                FIXINH     27
      END                                                                    DTRIM      33
```

```
      SUBROUTINE PTPLOT(NX,NY)                                          PTPLOT     2
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                  CYBTTT     1
     X     ISTR,IBUT,TMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77    3
      DIMENSION NAMES(4),PNAME(6),MSSG(2),LX(2),NVXY(2)                 5OCT       58
      DATA NAMES/2HCL,6HDELTAC,5HALPHA,2HCD/,MSSG(2)/10HHEN ACCEPT/     PTPLOT     5
     1,LX/10HPICK X,  T,10HPICK Y,  T/                                  5OCT       59
      YOFF=.5                                                           PTPLOT     7
      YV=YV=0.                                                          PTPLOT     8
      DO 20 I=1,4                                                       PTPLOT     9
      YV=YV-YOFF                                                        PTPLOT     10
   20 CALL ANNOTA(XV,YV,NAMES(I),10,PNAME(I),ISIN,I,-0,-0)              PTPLOT     11
      CALL ANNOTA(XV,-3.,6HACCEPT,6,PNAME(5),IBUT,23,-0,-0)             PTPLOT     12
      DO 50 J=1,2                                                       PTPLOT     13
      MSSG(1)=LX(J)                                                     5OCT       60
      CALL ANNOTA(XV,0.,MSSG,20,PNAME(6),IGN,-0,-0,-0)                  PTPLOT     15
   21 CALL GIBUT(0,NCON,IDDTR,IDDCR)                                    PTPLOT     16
      IF(IDDCR.EQ.23)GO TO 23                                           PTPLOT     17
      GO TO 21                                                          PTPLOT     18
   23 CALL GIFIE(NCON,IDDTR,IDDCR)                                      PTPLOT     19
      IF(IDDTR.NE.0)GO TO 25                                            PTPLOT     20
      CALL ERROR4(XV,-4.,37HFIRST PICK PLOT VARIABLE, THEN ACCEPT  ,37, PTPLOT     21
     16HPTPLOT,6HINFORM)                                                PTPLOT     22
      GO TO 21                                                          PTPLOT     23
   25 NVXY(J)=IDDCR                                                     PTPLOT     24
      CALL GERAS(PNAME(6))                                              PTPLOT     25
   50 CONTINUE                                                          PTPLOT     26
      NX=NVXY(1)                                                        PTPLOT     27
      NY=NVXY(2)                                                        PTPLOT     28
      CALL TERASE(PNAME,5)                                              PTPLOT     29
      END                                                               PTPLOT     30
```

```
      SUBROUTINE TDISP(I300,IBEAD,IVALU)                                TDISP     2
      DIMENSION NAMES(4)                                                TDISP     3
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/   2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG     1
C     J=N*2**1?  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/   3
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/  2
      COMMON/TDIS/ITRIM1,PRNAME,NX,NY,AA(8)                             50CT     61
      DATA NAMES/2HCL,6HDELTAC,6HALPHA,2HCD/                            TDISP     8
      J=10*2**18                                                        TDISP     9
      CALL DMGET(ICOM7+J,IBEAD,NREC)                                    TDISP    10
      NPTS=NREC(5)                                                      TDISP    11
      CALL DMGT?D(NPTS,IREC)                                            22JUNE   39
      CALL DMGT?D(14,IPLT)                                              22JUNE   40
      CALL DMGT?D(12,ICRV)                                              22JUNE   41
      CALL DMGT?D(7,IPG)                                                22JUNE   42
      CALL DMSET(ICOM1+5,IPLT,IPG)                                      TDISP    16
      CALL DMGT?D(10,ITITL)                                             22JUNE   43
      CALL DMSET(ICOM1+6,IPLT,ITITL)                                    TDISP    18
      CALL DMGT?D(7,IPAX)                                               22JUNE   44
      CALL DMSET(ICOM1+7,IPLT,IPAX)                                     TDISP    20
      IF(IBOC.EQ.IBEAD)CALL DMGT?D(8,IPAXSC)                            3FEB     10
      CALL DMSET(ICOM1+8,IPLT,IPAXSC)                                   TDISP    22
      CALL DMGET(ICOM1+9,IPLTPTR,IPM)                                   TDISP    23
      CALL DMSET(ICOM1+9,IPLT,IPM)                                      TDISP    24
      CALL DMSET(ICOM1+10,IPLT,IPLT)                                    TDISP    25
      CALL DMSET(ICOM1+11,IPLT,ICRV)                                    TDISP    26
      CALL DMSET(ICOM1+12,IPLT,1)                                       TDISP
      CALL DMSET(ICOM1+13,IPLT,NREC(7))                                 TDISP
      CALL DMSET(ICOM1+14,IPLT,NREC(6))                                 TDISP    29
      CALL DMSET(ICOM1+3,ICRV,PRNAME)                                   22JUNE   46
      CALL DMSET(ICOM1+5,ICRV,NPTS)                                     TDISP    30
      CALL DMSET(ICOM1+6,ICRV,1)                                        TDISP    31
      CALL DMSET(ICOM1+7,ICRV,IREC)                                     TDISP    32
      CALL DMSET(ICOM1+8,ICRV,NREC(NX))                                 TDISP    33
      CALL DMSET(ICOM1+9,ICRV,NREC(NY))                                 TDISP    34
      CALL DMSET(ICOM1+10,ICRV,1HX)                                     TDISP    35
      CALL DMSET(ICOM1+12,ICRV,6HACTIVE)                                22JUNE   47
      JJ=NPTS*2**18                                                     22JUNE   48
      CALL DMGET(ICOM7+JJ,NREC(NX),X)                                   22JUNE   49
      CALL DMGET(ICOM7+JJ,NREC(NY),Y)                                   22JUNE   50
      IF(IBOC.EQ.IBEAD)CALL ASCALE(X,Y,NPTS)                            3FEB     11
      CALL DMGET(ICOM1+5,IPLTPTR,IPPG)                                  TDISP    36
      JJ=7*2**18                                                        TDISP    37
      CALL DMGET(ICOM7+JJ,IPPG,NREC)                                    TDISP    38
      CALL DMSET(ICOM7+JJ,IPG,NREC)                                     TDISP    39
      CALL DMSET(ICOM1+3,IPG,ITITL)                                     TDISP    40
      CALL DMSET(ICOM1+4,IPG,IPAX)                                      TDISP    41
      CALL DMSET(ICOM1+4,IPAX,IPM)                                      TDISP    42
      CALL DMSET(ICOM1+5,IPAX,NAMES(NX))                                TDISP    43
      CALL DMSET(ICOM1+6,IPAX,1)                                        TDISP    44
      CALL DMSET(ICOM1+7,IPAX,NAMES(NY))                                TDISP    45
      IF(IBOC.NE.IBEAD)GO TO 40                                         3FEB     12
      JJ=8*2**18                                                        T30AUG    6
      CALL DMSET(ICOM7+JJ,IPAXSC,AA)                                    T30AUG    7
   40 CONTINUE                                                          3FEB     13
      CALL DMSET(ICOM1+2,ITITL,10)                                      TDISP    50
      CALL DMSET(ICOM1+7,ITITL,10HTRIM PLOT )                           TDISP    51
```

```
                      JJ=NPTS*2**18                                          TDISP      52
                      DO 51 I=1,NPTS                                         TDISP      53
60          51        NREC(I)=1                                              TDISP      54
                      CALL DMSET(ICOM7+JJ,IREC,NREC)                         TDISP      55
                      IF(IRCC.EQ.IREAD)GO TO 20                              TDISP      56
                      CALL LINK(ITRIM1,IPLT,3,1)                             TDISP      57
                      RETURN                                                 TDISP      58
65          20        CONTINUE                                               TDISP      59
                      ITRIM1=IPLT                                            TDISP      60
                      END                                                    TDISP      61
```

```
      SUBROUTINE ASCALE(XA,YA,NPA)                                      ASCALE     2
      COMMON/TOIS/ITRIM1,PRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,      T3OAUG    21
     1XDATMAX,YDATMIN,YDATMAX                                           T3OAUG    22
      DIMENSION XA(1),YA(1)                                             ASCALE     4
      DATA XL/9./,YL/6./                                                ASCALE     5
      XMIN=AMIN(XA,NPA)                                                 ASCALE     6
      XMAX=AMAX(XA,NPA)                                                 ASCALE     7
      YMIN=AMIN(YA,NPA)                                                 ASCALE     8
      YMAX=AMAX(YA,NPA)                                                 ASCALE     9
      XDATMIN=XMIN                                                      T3OAUG     8
      XDATMAX=XMAX                                                      T3OAUG     9
      YDATMIN=YMIN                                                      T3OAUG    10
      YDATMAX=YMAX                                                      T3OAUG    11
      DDX=.1*(XMAX-XMIN)                                                ASCALE    10
      DDY=.1*(YMAX-YMIN)                                                ASCALE    11
      XMIN=XMIN-DDX                                                     ASCALE    12
      YMIN=YMIN-DDY                                                     ASCALE    13
      XMAX=XMAX+DDX                                                     ASCALE    14
      YMAX=YMAX+DDY                                                     ASCALE    15
      DELX=(XMAX-XMIN)/XL                                               ASCALE    16
      DELY=(YMAX-YMIN)/YL                                               ASCALE    17
      END                                                               ASCALE    18
```

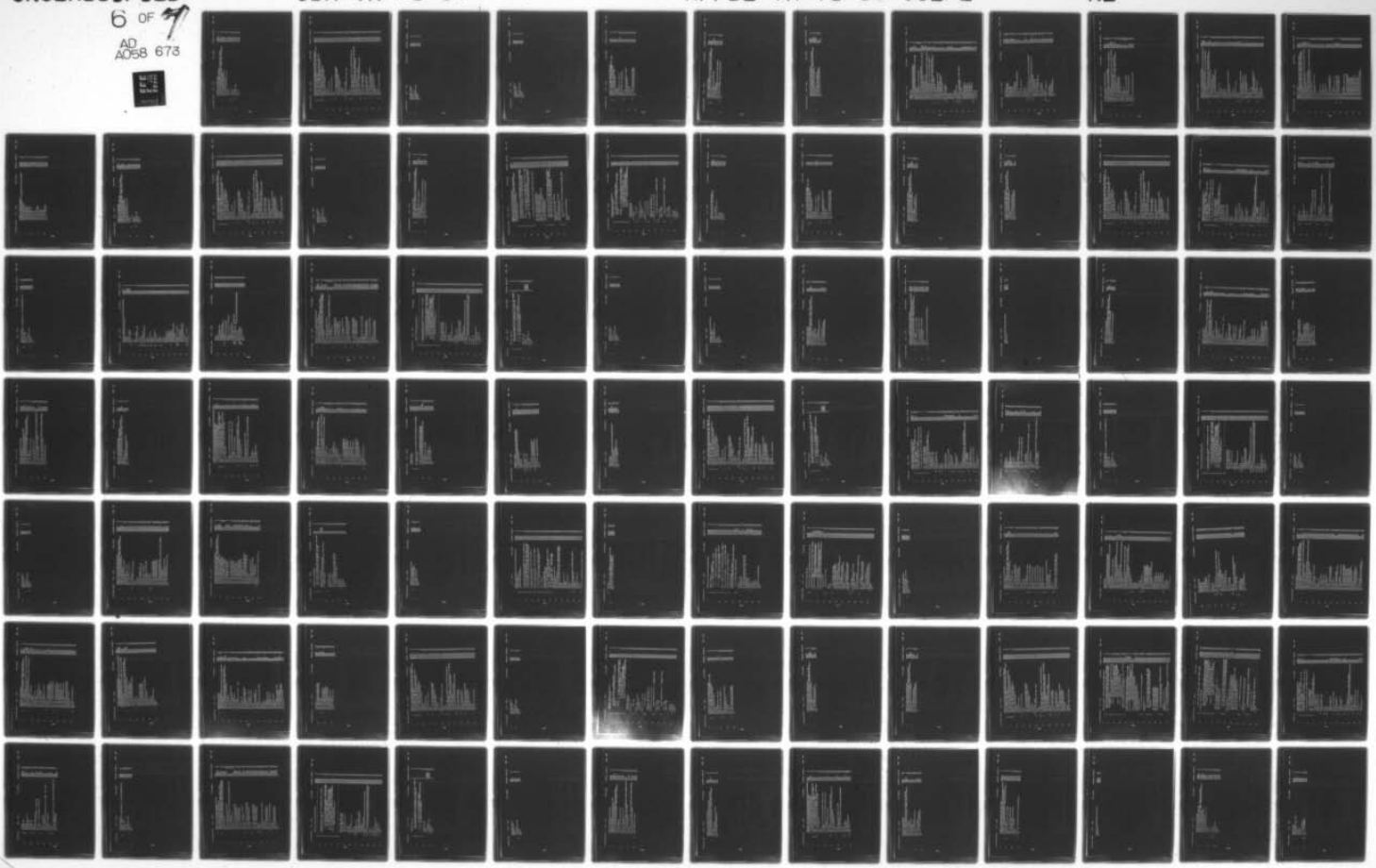
```
      SUBROUTINE SPLICE2(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU
C     IN WORD JMATCH OF EACH BEAD.
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT
C     POINTER TO PREVIOUS BEAD IN WORD JPREV
C     POINTER TO NEXT BEAD IN WORD JNEXT
      DATA(COMP/01000(000000CR/
      NEXT=IBD1
      IPREV=IBD1
      IF(IBD1.NE.0)GOTO10
      WRITE (6,5)
    5 FORMAT(13H IBD1 IS NULL )
      IS=2
      RETURN
   10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)
      IF(ICKVALU.EQ.IVALU) GOTO15
      IF(NEXTAD.EQ.0)GOTO12
      IPREV=NEXT
      NEXT=NEXTAD
      GO TO 10
   12 WRITE (6,6) IVALU
    6 FORMAT(19H NO MATCH FOR IVALU ,O20)
      IS=2
      RETURN
   15 CONTINUE
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL
      IF(NEXT.EQ.IBD1)GO TO 25
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)
      IF(NEXTAD.EQ.0)GOTO22
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)
   22 CONTINUE
      CALL SUBR(IBD1,NEXT,IVALU)
      IF(IS.EQ.1)CALL DMRLBD(NEXT)
      RETURN
   25 CALL SUBR(IBD1,IBD1,IVALU)
      IF(NEXTAD.EQ.0)GO TO 40
      NEXT=NEXTAD
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)
      CALL SUBR(IBD1,NEXT,IVALU)
      IF(IS.EQ.1)CALL DMRLBD(NEXT)
      IF(NEXTAD.EQ.0)GO TO 40
      NEXT=NEXTAD
      GO TO 30
   40 CONTINUE
      IF(IS.EQ.1)CALL DMRLBD(IBD1)
      END
```

```
      SUBROUTINE RLSPLT(IBEAD1,IBEAD,IVALU)                              RLSPLT     2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG      1
C     J=N*2**19  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/   2
      EXTERNAL RLSCRV                                                    RLSPLT     5
      CALL DMSET(ICOM1+11,IBEAD,ICRV)                                    50CT     108
      CALL SPLICE2(ICRV,RLSCRV,2,0,1,2,1)                                RLSPLT    11
      END                                                                RLSPLT    17
```

AD-A058 673 DAYTON UNIV OHIO RESEARCH INST F/G 9/2
PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). VOLUME II--ETC(U)
MAY 78 J G JENSEN F33615-76-C-3076
UNCLASSIFIED UDR-TR-78-34 AFFDL-TR-78-55-VOL-2 NL

6 OF 9

AD A058 673



```
      SUBROUTINE ITRIMR(IBEAD)                                          ITRIMR     2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/    2
      COMMON/DISKT/I90D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG         30AUG      1
C     J=N*2**19  DMSET(ICOM7+J,I90A,DATA)  STORES DATA(N) IN IBOA(1)    /DISKT/    3
      COMMON/OPTION/IOPTION,ISHINE                                      LASTBUX    1
      CALL DMGET(ICOM1+2,IBEAD,ITRIM)                                   ITRIMR     5
      IF(ITRIM.EQ.0)GO TO 10                                            ITRIMR     6
      IF (IOPTION.NE.4HTRIM) GO TO 10                                   3AUG      16
      INEXT=2                                                           ITRIMR     7
      IPREV=4                                                           ITRIMR     8
C                                                                       ITRIMR     9
C     TRIM MODE                                                         ITRIMR    10
      RETURN                                                            ITRIMR    11
   10 CONTINUE                                                          ITRIMR    12
C     CROSSPLOT MODE                                                    ITRIMR    13
      INEXT=1                                                           ITRIMR    14
      IPREV=3                                                           ITRIMR    15
      END                                                               ITRIMR    16
```

```
      SUBROUTINE SPLICE(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)                  SPLICE     2
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU            SPLICE     3
C     IN WORD JMATCH OF EACH BEAD.                                              SPLICE     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND                     SPLICE     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                            SPLICE     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                        SPLICE     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                                    SPLICE     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                        SPLICE     9
      DATA(COMP/8100000000008/                                                  SPLICE    10
      NEXT=IBD1                                                                 SPLICE    11
      IPREV=IBD1                                                                SPLICE    12
      IF(IBD1.NE.0)GOTO10                                                       SPLICE    13
      PRINT                                                                     SPLICE    14
    5 FORMAT(17H IBD1 IS NULL )                                                 SPLICE    15
      IS=2                                                                      SPLICE    16
      RETURN                                                                    SPLICE    17
   10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                                     SPLICE    18
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                       SPLICE    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                               SPLICE    20
      IF(NEXTAD.EQ.0)GOTO12                                                     SPLICE    21
      IPREV=NEXT                                                                SPLICE    22
      NEXT=NEXTAD                                                               SPLICE    23
      GO TO 10                                                                  SPLICE    24
   12 WRITE (6,6) IVALU                                                         CYB777    21
    6 FORMAT(19H NO MATCH FOR IVALU ,020)                                       SPLICE    26
      IS=2                                                                      SPLICE    27
      RETURN                                                                    SPLICE
   15 CONTINUE                                                                  SPLICE
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL                 SPLICE    30
      IF(NEXT.EQ.IBD1)GO TO 25                                                  SPLICE    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE             SPLICE    32
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                           SPLICE    33
      IF(NEXTAD.EQ.0)GOTO22                                                     SPLICE    34
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                       SPLICE    35
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                          SPLICE    36
   22 CONTINUE                                                                  SPLICE    37
      CALL SUBR(IBD1,NEXT,IVALU)                                                SPLICE    38
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                              SPLICE    39
      RETURN                                                                    SPLICE    40
   25 CALL SUBR(IBD1,IBD1,IVALU)                                                SPLICE    41
      IF(NEXTAD.EQ.0)GO TO 40                                                   SPLICE    42
      NEXT=NEXTAD                                                               SPLICE    43
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                       SPLICE    44
      CALL SUBR(IBD1,NEXT,IVALU)                                                SPLICE    45
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                              SPLICE    46
      IF(NEXTAD.EQ.0)GO TO 40                                                   SPLICE    47
      NEXT=NEXTAD                                                               SPLICE    48
      GO TO 30                                                                  SPLICE    49
   40 CONTINUE                                                                  SPLICE    50
      IF(IS.EQ.1)CALL DMRLBD(IBD1)                                              SPLICE    51
      END                                                                       SPLICE    52
```

```
1                FUNCTION AMAX(A,N)                                    AMAX      2
                 DIMENSION A(N)                                        AMAX      3
                 V=A(1)                                                AMAX      4
                 DO 10 I=2,N                                           AMAX      5
5                IF(A(I).GT.V)V=A(I)                                   AMAX      6
              10 CONTINUE                                              AMAX      7
                 AMAX=V                                                AMAX      8
                 END                                                   AMAX      9
```

```
      FUNCTION AMIN(A,N)                                        AMIN    2
      DIMENSION A(N)                                            AMIN    3
      V=A(1)                                                    AMIN    4
      DO 10 I=2,N                                               AMIN    5
      IF(A(I).LT.V)V=A(I)                                       AMIN    6
   10 CONTINUE                                                  AMIN    7
      AMIN=V                                                    AMIN    8
      END                                                       AMIN    9
```

```
      SUBROUTINE LINK(IBD1,IBONEW,JPREV,JNEXT)                          LINK        2
C     ATTACHES LINK IBONEW TO STRING BEGINNING WITH IBD1                LINK        3
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                            LINK        4
C     POINTER TO NEXT BEAD IN WORD JNEXT                                LINK        5
      DATA ICOMP/8100000000008/                                         LINK        6
      NEXT=IBD1                                                         LINK        7
      JPREV=IBD1                                                        LINK        8
      IF(IBD1*IBONEW.NE. 0)GO TO 10                                     LINK        9
      WRITE (6,5)                                                       CY8777     24
    5 FORMAT(20H NULL BEAD ADDRESSES )                                  LINK       11
      STOP                                                              LINK       12
   10 CONTINUE                                                          LINK       13
      CALL GMGET(ICOMP+JNEXT,NEXT,NEXTAD)                               LINK       14
      IF(NEXTAD.EQ.0)GO TO 20                                           LINK       15
   15 JPREV=NEXTAD                                                      LINK       16
      NEXT=NEXTAD                                                       LINK       17
      GO TO 10                                                          LINK       18
   20 CALL GMSET(ICOMP+JNEXT,NEXT,IBONEW)                               LINK       19
      CALL GMSET(ICOMP+JPREV,IBONEW,IPREV)                              LINK       20
   20 END                                                               LINK       21
```

```
1                SUBROUTINE RLSPM(IPLTPTR)                                  RLSPM      2
                 C  RELEASES P-MGROUP BEGINNING WITH PLOT IPLTPTR.ITS P-M BEAD  RLSPM      3
                 C  IS NOT RELEASED BUT THE TITLE, P-G BEAD,AXIS AND AXISSCALEBEADS RLSPM      4
                 C  ARE RELEASED.                                              RLSPM      5
5                COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                         /ITRIMT/   2
                 EXTERNAL RLSPLT                                            RLSPM      6
                 CALL ITRIMR(IPLTPTR)                                       RLSPM      7
                 CALL SPLICE(IPLTPTR,RLSPLT,IPREV,3,INEXT,IPREV,1)          RLSPM      8
                 WRITE (6,10) IPLTPTR                                       CYBTTY     26
10               10 FORMAT(1H0,*P-M GROUP IPLTPTR *,I10,*RELEASED*)         RLSPM      10
                 END                                                        RLSPM      11
```

```
1                SUBROUTINE RLSCRV(IBEAD1,IBEAD,IVALU)                          RLSCRV     2
                 COMMON/DISKT/IBD1,ICOM1,ICOM2,XVAL,XX(20),YY(20),IC,NDEG       30AUG      1
           C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/    3
                 COMMON/POINTS/X(248),Y(248),NREC(248)                          /POINTS/   2
5                JJ=12*2**18                                                    RLSCRV     5
                 CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                RLSCRV     6
                 CALL DMRLBD(NREC(7),NREC(8),NREC(9))                           TOM       36
                 IF(NREC(11).NE.0)CALL DMRLBD(NREC(11))                         TOM       37
                 END                                                            RLSCRV     8
```

```
      OVERLAY(14,0)                                                    CROSS      2
      PROGRAM CROSS                                                    CROSS      3
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                 CYBTTT     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                6JULY77    3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1   10MAY77    61
     1,IDELTF,IDELBD(2)                                               FLCAD2     5
      DIMENSION IDA(10)                                                2NOV       110
      COMMON/PXVAL/PPVALU(10),IPPV,PXNAM,NCRVS2                        16NOV      15
      EXTERNAL PAPSEL                                                  2NOV       112
      COMMON/PLOTCOM/IFIRST,IPLTFL1,NCPC,ICALBD(100)                   25AUG      41
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10) /CROSTB/  2
     1,EMACH,PVALU(10),PNAME(10),IPP(10)                               /CROSTB/   3
      COMMON/MULTKS/NKS,AXK(20)                                        13SEPT     30
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                               /ITRIMT/   2
      COMMON/BEADS/IEGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/   2
      EXTERNAL SUBCPLT,RLSPM                                           CROSS      8
      COMMON/TDIS/ITRIM1,FRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,     T30AUG     21
     1XDATMAX,YDATMIN,YDATMAX                                          T30AUG     22
      COMMON/DISKT/IBC1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG         30AUG      1
C     J=N*2**14  OMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)   /DISKT/    3
      DATA XL/0./,YL/0./,BIG/1.0E33/                                   22JUNE     54
      IDELBD(1)=IDELBD(2)=0                                            13SEPT     31
C     RESTORE IPLTPTR                                                  CROSS      16
      IF(IPLTPTR.EQ.ICROSS1)IPLTPTR=ICPSAVE                            CROSS      17
      IF(ICROSS1.EQ.0)GO TO 20                                         CROSS      18
C     IF PRIOR CROSSPLOT STORED,RELEASE IT                             CROSS      19
      CALL LMGET(ICOM1+10,ICROSS1,IPLT)                                25AUG
      IF(IPLT.NE.99)GO TO 19                                           25AUG
      GO TO 20                                                         25AUG      44
   19 CALL ITRIMR (ICROSS1)                                            25AUG      45
      CALL RLSPM(ICROSS1)                                              TOM        23
   20 CONTINUE                                                         CROSS      22
      XMIN=BIG                                                         22JUNE     56
      XMAX=-BIG                                                        22JUNE     57
      YMIN=BIG                                                         22JUNE     58
      YMAX=-BIG                                                        22JUNE     59
      NCRVS=0                                                          CROSS      23
      DO 15 I=1,10                                                     CROSS      24
   15   NCVAL(I)=0                                                     CROSS      25
      CALL ITRIMR(IPLTPTR)                                             CROSS      26
 1603 CONTINUE                                                         2NOV       145
      CALL SPLICE (IPLTPTR,SUBCPLT,3,0,INEXT,IPREV,0)                  3AUG       6
  100 FORMAT(1H0,I5,5X,A10,F10.5)                                      CROSS      29
      DO 300 J=1,NCRVS                                                 CROSS      30
      K=NCVAL(J)                                                       CROSS      31
      XMN=AMIN(XM(1,J),K)                                              22JUNE     60
      IF(XMN.LT.XMIN)XMIN=XMN                                          22JUNE     61
      XMX=AMAX(XM(1,J),K)                                              22JUNE     62
      IF(XMX.GT.XMAX)XMAX=XMX                                          22JUNE     63
      YMN=AMIN(YM(1,J),K)                                              22JUNE     64
      IF(YMN.LT.YMIN)YMIN=YMN                                          22JUNE     65
      YMX=AMAX(YM(1,J),K)                                              22JUNE     66
      IF(YMX.GT.YMAX)YMAX=YMX                                          22JUNE     67
  300 CONTINUE                                                         CROSS      37
      XDATMIN=YMIN                                                     T30AUG     14
      XDATMAX=XMAX                                                     T30AUG     15
      YDATMIN=YMIN                                                     T30AUG     16
```

```
                YMATMAX=YMAX                                                    T30AUG     17
                DDX=.1*(XMAX-XMIN)                                              22JUNE     68
60              DDY=.1*(YMAX-YMIN)                                              22JUNE     69
                XMIN=XMIN-DDX                                                   22JUNE     70
                YMIN=YMIN-DDY                                                   22JUNE     71
                XMAX=XMAX+DDX                                                   22JUNE     72
                YMAX=YMAX+DDY                                                   22JUNE     73
65              DELX=(XMAX-XMIN)/XL                                             22JUNE     74
                DELY=(YMAX-YMIN)/YL                                             22JUNE     75
                IF (JT.EQ.4) YNAME=10HCOORDINATE                                3FEB       21
                IVC=1HX                                                         15FEB       9
                CALL CROSSD                                                     CROSS      38
70              IF (JT.NE.4) GO TO 500                                          3FEB       22
                JT=0                                                            3FEB       23
                YMATMIN=BIG                                                     3FEB       24
                YMATMAX=-BIG                                                    3FEB       25
            400 CONTINUE                                                        20SEPT     115
75              CALL ORIGIN(-5.,5.)                                             20SEPT     116
            402 CALL ANNOTA (0.,0.,26HDELTA CROSS PLOT SELECTION,26,            3FEB       26
               1    IDA1,IBUT,7,-0,-0)                                          3FEB       27
                CALL ANNOTA(0.,-1.,8HCONTINUE,8,IDA2,IBUT,8,-0,-0)              20SEPT     118
            405 CALL GIBUT(C,NCON,IDDT,IDDC)                                    20SEPT     119
80              CALL GIBAS(IDA1,IDA2)                                           20SEPT     120
                IF(IDDC.EQ.8)GO TO 500                                          20SEPT     121
                IF(IDDC.NE.7)GO TO 402                                          20SEPT     122
                CALL DELPIC(ICROSS1,ICELBD)                                     13SEPT      39
                JT=JT+1                                                         3FEB       28
85              CALL DELBD(ICELB)                                               13SEPT      40
                CALL SKALBD(ICELB)                                              3FEB       29
                GO TO 402                                                       20SEPT     123
            500 CONTINUE                                                        13SEPT      41
              C     DISPLAY CROSSPLOTS                                          CROSS      41
90            C                                                                 CROSS      42
              C     SAVE IPLTPTR                                                CROSS      43
                    ICRSAVE=IPLTPTR                                             CROSS      44
                    IPLTPTR=ICROSS1                                             CROSS      45
                    CALL AETSKC(7HFRSTPL2)                                      10MAY77    60
95                  END                                                         CROSS      47
```

```
      SUBROUTINE SUBCPLT(IBEAD1,IBDE,IVALU)                                  CROSS      46
      EXTERNAL SUBCCRV                                                       CROSS      49
      COMMON/PXVAL/PPVALU(10),IPPV,PXNAM,NCRVS2                              16NOV      19
      COMMON/CROSTBL/ICROS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1          10MAY77    61
     1,IDELTF,IDFLAG(2)                                                      FLCA02     5
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)      /CROSTB/   2
     1,FMACH,PVALU(10),PNAME(10),IRP(10)                                     /CROSTB/   3
      COMMON/DISKT/IBD1,ICOM1,ICOM2,XVAL,XX(20),YY(20),IC,NDEG               30AUG      1
C     J=N*2**15  DMSET(ICOM2+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/    3
      COMMON/MULTKS/NKS,AXK(20)                                              13SEPT     42
      CALL DMGET(ICOM1+11,IBDE,IBEAD)                                        CROSS      53
      CALL DMGET(ICOM1+14,IBDE,FMACH)                                        CROSS      54
      IF(NKS.LE.1)GO TO 10                                                   13SEPT     43
      IBEADP=IBEAD                                                           2NOV       147
      IF(IPPV.EQ.1)GO TO 5                                                   2NOV       148
      DO 4 I=2,IPPV                                                          2NOV       149
      CALL DMGET(ICOM1+1,IBEADP,IBEADN)                                      2NOV       150
    4 IBEADP=IBEADN                                                          2NOV       151
    5 CALL SUBCCRV(IBEADP,IBEADP,IVALU)                                      2NOV       152
      RETURN                                                                 13SEPT     45
   10 CONTINUE                                                               13SEPT     46
      CALL SPLICE2(IBEAD,SUBCCRV,2,0,1,2,0)                                  CROSS      55
      END                                                                    CROSS      56
```

```
      SUBROUTINE SUBCCRV(IBEAD1,IBEAD,IVALU)                                CROSS      57
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)     /CROSTB/     2
     1,FMACH,PVALU(10),PNAME(10),IRP(10)                                    /CROSTB/     3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1        10MAY77    61
     1,IDELTF,IFELBC(2)                                                     FLCAD2     5
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG              30AUG      1
C     J=N*2**18  CMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/     3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                 /POINTS/     2
      COMMON/MULTKS/NKS,AXK(20)                                             13SEPT     47
      CALL CMGET(ICOM1+12,IBEAD,ISTAT)                                      CROSS      62
      IF(ISTAT.EQ.8HINACTIVE)RETURN                                         CROSS      63
      CALL FITBLK(IBEAD,NF,NDEG,IFIT,IDVAR)                                 16AUG76    11
      XK=AXK(1)                                                             13SEPT     48
      IVC=IDVAR                                                             20SEPT    124
      IF(NKS.LE.1)GO TO 60                                                  13SEPT     49
   10   NCRVS=NKS                                                          13SEPT     50
      DO 18 J=1,NKS                                                         13SEPT     51
      XK=AXK(J)                                                             13SEPT     52
      PNAME(J)=1HK                                                          13SEPT     53
      PVALU(J)=XK                                                           13SEPT     54
      ISYM(J)=J                                                             20SEPT    125
      YV=YVA(NP,NDEG,IFIT,JT,IDVAR,IER,XK)                                  13SEPT     55
      NCVAL(J)=NCVAL(J)+1                                                   13SEPT     56
      I=NCVAL(J)                                                            13SEPT     57
      IRP(I)=1                                                              13SEPT     58
      XM(I,J)=FMACH                                                         13SEPT     59
      YM(I,J)=1.0                                                           29APR
      IF(IER.LE.1)YM(I,J)=YV                                                13SEPT
   18 CONTINUE                                                              13SEPT     62
      RETURN                                                                13SEPT     63
   60 CONTINUE                                                              13SEPT     64
      YV=YVA(NP,NDEG,IFIT,JT,IDVAR,IER,XK)                                  16AUG76    12
   25 IF (IER.GT.1) RETURN                                                  FIX2       3
      CALL CMGET(ICOM1+4,IBEAD,PVAL)                                        CROSS      79
      IF(NCRVS.EQ.0)GO TO 40                                                CROSS      80
      CALL SEEK(PVAL,PVALU,NCRVS,ICV)                                       CROSS      81
      IF(ICV.NE.0)GO TO 50                                                  CROSS      82
   40 CONTINUE                                                              CROSS      83
      NCRVS=NCRVS+1                                                         CROSS      84
      ICV=NCRVS                                                             CROSS      85
      PVALU(ICV)=PVAL                                                       CROSS      86
      CALL CMSET(ICOM1+3,IBEAD,PNAME(ICV))                                  CROSS      87
      CALL CMGET(ICOM1+6,IBEAD,ISY)                                         CROSS      88
      ISYM(ICV)=ISY                                                         CROSS      89
   50 NCVAL(ICV)=NCVAL(ICV)+1                                               CROSS      90
      I=NCVAL(ICV)                                                          CROSS      91
      IRP(I)=1                                                              CROSS      92
      XM(I,ICV)=FMACH                                                       CROSS      93
      YM(I,ICV)=YV                                                          CROSS      94
      END                                                                   CROSS     103
```

```
      SUBROUTINE CROSBD                                                 CROSS   104
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG     1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/   3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1    10MAY77  61
     1,TDELTF,IDELBD(2)                                                 FLOAD2    5
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10) /CROSTB/   2
     1,EMACH,PVALU(10),PNAME(10),IRP(10)                                /CROSTB/   3
      COMMON /TTEMP/TEMP(10)                                            /TTEMP/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      COMMON /BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/   2
      COMMON/TDIS/ITRIM1,FRNAME,NX,NY,AA(8)                             T30AUG   18
      CALL DMGTBD(14,ICROSS1)                                           CROSS   111
      JJ=14*2**18                                                       CROSS   112
      CALL DMGET(ICOM7+JJ,IPLTPTR,NREC)                                 CROSS   113
      CALL DMGTBD(7,IPG)                                                CROSS   114
      NREC(1)=6                                                         CROSS   115
      NREC(2)=6                                                         3AUG      8
      NREC(5)=IPG                                                       CROSS   116
      CALL DMGTBD(10,ITITL)                                             CROSS   117
      NREC(6)=ITITL                                                     CROSS   118
      CALL DMGTBD(7,IAXIS)                                              CROSS   119
      NREC(7)=IAXIS                                                     CROSS   120
      CALL DMGTBD(8,IAXISC)                                             TOM      25
      NREC(8)=IAXISC                                                    CROSS   122
      LL=8*2**18                                                        20SEPT  126
      CALL DMSET (ICOM7+LL,IAXISC,AA)                                   20SEPT  127
      CALL DMGET(ICOM1+9,IPLTPTR,IPM)                                   CROSS   123
      NREC(9)=IPM                                                       CROSS   124
      NREC(10)=ICROSS1                                                  CROSS   125
      IF(JT.EQ.3.OR.JT.EQ.5)GO TO 12                                    FIXMIN    4
      NREC(13)=1HK                                                      FIXMIN    5
   12 CALL DMSET(ICOM7+JJ,ICROSS1,NREC)                                 FIXMIN    6
      IF(JT.EQ.3.OR.JT.EQ.5)GO TO 14                                    FIXMIN    7
      CALL DMSET(ICOM1+14,ICROSS1,XK)                                   FIXMIN    8
   14 CONTINUE                                                          FIXMIN    9
      ICRV1=0                                                           CROSS   127
      CALL SAVEC(ICRV1)                                                 CROSS   128
      CALL DMSET(ICOM1+11,ICROSS1,ICRV1)                                CROSS   129
      JJ=7*2**18                                                        CROSS   130
      CALL DMGET(ICOM1+5,IPLTPTR,IPGOLD)                                CROSS   131
      CALL DMGET(ICOM7+JJ,IPGOLD,TEMP)                                  CROSS   132
      CALL DMSET(ICOM7+JJ,IPG,TEMP)                                     CROSS   133
      CALL DMSET(ICOM1+3,IPG,ITITL)                                     CROSS   134
      CALL DMSET(ICOM1+4,IPG,IAXIS)                                     CROSS   135
      CALL DMSET(ICOM1+4,IAXIS,IPM)                                     CROSS   136
      CALL DMSET(ICOM1+5,IAXIS,4HMACH)                                  CROSS   137
      CALL DMSET(ICOM1+6,IAXIS,1)                                       CROSS   138
      CALL DMSET(ICOM1+7,IAXIS,YNAME)                                   CROSS   139
      CALL DMSET(ICOM1+2,ITITL,20)                                      CROSS   144
      CALL DMSET(ICOM1+3,ITITL,CTYPE(JT))                               CROSS   145
      CALL DMSET(ICOM1+4,ITITL ,10H CROSSPLOT)                          CROSS   146
      END                                                               CROSS   147
```

```
      SUBROUTINE ASCALE(XA,YA,NPA)                                      ASCALE      2
      COMMON/TDIS/ITRIM1,PRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,      T30AUG      21
     1XDATMAX,YDATMIN,YDATMAX                                           T30AUG      22
      DIMENSION XA(1),YA(1)                                             ASCALE      4
      DATA XL/9./,YL/6./                                                ASCALE      5
      XMIN=AMIN(XA,NPA)                                                 ASCALE      6
      XMAX=AMAX(XA,NPA)                                                 ASCALE      7
      YMIN=AMIN(YA,NPA)                                                 ASCALE      8
      YMAX=AMAX(YA,NPA)                                                 ASCALE      9
      XDATMIN=XMIN                                                      T30AUG      8
      XDATMAX=XMAX                                                      T30AUG      9
      YDATMIN=YMIN                                                      T30AUG      10
      YDATMAX=YMAX                                                      T30AUG      11
      DDX=.1*(XMAX-XMIN)                                                ASCALE      10
      DDY=.1*(YMAX-YMIN)                                                ASCALE      11
      XMIN=XMIN-DDX                                                     ASCALE      12
      YMIN=YMIN-DDY                                                     ASCALE      13
      XMAX=XMAX+DDX                                                     ASCALE      14
      YMAX=YMAX+DDY                                                     ASCALE      15
      DELX=(XMAX-XMIN)/XL                                               ASCALE      16
      DELY=(YMAX-YMIN)/YL                                               ASCALE      17
      END                                                               ASCALE      18
```

```
      SUBROUTINE ITRIMR(IBEAD)                                          ITRIMR     2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      COMMON/OPTION/ICPTION,ISHINE                                      LASTBUX    1
      CALL DMGET(ICOM1+2,IBEAD,ITRIM)                                   ITRIMR     5
      IF(ITRIM.EQ.0)GO TO 10                                            ITRIMR     6
      IF (ICPTION.NE.4HTRIM) GO TO 10                                   3AUG      16
      INEXT=2                                                           ITRIMR     7
      IPREV=4                                                           ITRIMR     8
C                                                                       ITRIMR     9
C     TRIM MODE                                                         ITRIMR    10
      RETURN                                                            ITRIMR    11
   10 CONTINUE                                                          ITRIMR    12
C     CROSSPLOT MODE                                                    ITRIMR    13
      INEXT=1                                                           ITRIMR    14
      IPREV=3                                                           ITRIMR    15
      END                                                               ITRIMR    16
```

```
      SUBROUTINE SPLICE2(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)          SPLICE2     2
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU     SPLICE2     3
C     IN WORD JMATCH OF EACH BEAD.                                       SPLICE2     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND              SPLICE2     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                     SPLICE2     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                 SPLICE2     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                             SPLICE2     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                 SPLICE2     9
      DATA ICOMP/6180000000000/                                          SPLICE2    10
      NEXT=IBD1                                                          SPLICE2    11
      IPREV=IBD1                                                         SPLICE2    12
      IF(IBD1.NE.0)GOTO10                                                SPLICE2    13
      WRITE (6,5)                                                        CYB777     22
    5 FORMAT(13H IBD1 IS NULL )                                          SPLICE2    15
      IS=2                                                               SPLICE2    16
      RETURN                                                             SPLICE2    17
   10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                              SPLICE2    18
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                SPLICE2    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                        SPLICE2    20
      IF(NEXTAD.EQ.0)GOTO12                                              SPLICE2    21
      IPREV=NEXT                                                         SPLICE2    22
      NEXT=NEXTAD                                                        SPLICE2    23
      GO TO 10                                                           SPLICE2    24
   12 WRITE (6,6) IVALU                                                  CYB777     25
    6 FORMAT(19H NO MATCH FOR IVALU ,020)                                SPLICE2    26
      IS=2                                                               SPLICE2    27
      RETURN                                                             SPLICE2    28
   15 CONTINUE                                                           SPLICE2    29
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL          SPLICE2    30
      IF(NEXT.EQ.IBD1)GO TO 25                                           SPLICE2    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE      SPLICE2    32
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                    SPLICE2    33
      IF(NEXTAD.EQ.0)GOTO22                                              SPLICE2    34
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                SPLICE2    35
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                   SPLICE2    36
   22 CONTINUE                                                           SPLICE2    37
      CALL SUBR(IBD1,NEXT,IVALU)                                         SPLICE2    38
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    39
      RETURN                                                             SPLICE2    40
   25 CALL SUBR(IBD1,IBD1,IVALU)                                         SPLICE2    41
      IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    42
      NEXT=NEXTAD                                                        SPLICE2    43
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                SPLICE2    44
      CALL SUBR(IBD1,NEXT,IVALU)                                         SPLICE2    45
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    46
      IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    47
      NEXT=NEXTAD                                                        SPLICE2    48
      GO TO 30                                                           SPLICE2    49
   40 CONTINUE                                                           SPLICE2    50
      IF(IS.EQ.1)CALL DMRLBD(IBD1)                                       SPLICE2    51
      END                                                                SPLICE2    52
```

```
1                FUNCTION AMAX(A,N)                                    AMAX       2
                 DIMENSION A(N)                                        AMAX       3
                 V=A(1)                                                AMAX       4
                 DO 10 I=2,N                                           AMAX       5
5                IF(A(I).GT.V)V=A(I)                                   AMAX       6
              10 CONTINUE                                              AMAX       7
                 AMAX=V                                                AMAX       8
                 END                                                   AMAX       9
```

```
1                SUBROUTINE RLSPM(IPLTPTR)                                  RLSPM      2
                 C  RELEASES P-MGROUP BEGINNING WITH PLOT IPLTPTR.ITS P-M BEAD  RLSPM      3
                 C  IS NOT RELEASED BUT THE TITLE, P-G BEAD,AXIS AND AXISSCALEBEADS RLSPM      4
                 C  ARE RELEASED.                                              RLSPM      5
5                COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                         /ITRIMT/   2
                 EXTERNAL RLSPLT                                            RLSPM      6
                 CALL ITRIMR(IPLTPTR)                                       RLSPM      7
                 CALL SPLICE(IPLTPTR,RLSPLT,IPREV,0,INEXT,IPREV,1)          RLSPM      8
                 WRITE (6,10) IPLTPTR                                       CY8777     26
10           10  FORMAT(1HO,*P-M GROUP IPLTPTR *,I10,*RELEASED*)            RLSPM      10
                 END                                                        RLSPM      11
```

```
      SUBROUTINE SORTCK2(ICRVPTR,IVK,NPOINTS)                          SORTCK2    2
      COMMON/DISKT/I801,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG         30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,I80A,DATA)  STORES DATA(N) IN IBDA(1)   /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                            /POINTS/   2
C     MODIFICATION OF SORTCK                                           SORTCK2    5
C                                                                      SORTCK2    6
C         THIS ROUTINE MOVES THE COORDINATE ARRAYS AND THE STATUS ARRAY SORTCK2    7
C         FROM THE DATA MANAGER FILE INTO CORE AND CHECKS TO SEE IF THE SORTCK2    8
C         POINTS HAVE TO BE SORTED. IF SO, THEN THEY ARE SORTED        SORTCK2    9
C                                                                      SORTCK2   10
C         ICRVPTR - I - THE POINTER TO THE CURVE HEAD OF THE CURVE WHICH SORTCK2  11
C                   IS TO BE EXAMINED FOR SORT ORDER.                  SORTCK2   12
C             IVK - I - THE INDEPENDENT VARIABLE SUBMITTED BY THE USER. SORTCK2   13
C         NPOINTS - O - THE NUMBER OF POINTS IN THE COORDINATE HEADS.  SORTCK2   14
C                                                                      SORTCK2   15
C                                                                      SORTCK2   16
C         MOVE THE COORDINATE ARRAYS AND THE STATUS ARRAY INTO CORE.   SORTCK2   17
C                                                                      SORTCK2   18
      CALL DMGET(ICOM1+5,ICRVPTR,NPOINTS)                              SORTCK2   19
      CALL DMGET(ICOM1+7,ICRVPTR,IRECPTR)                              SORTCK2   20
      CALL DMGET(ICOM1+8,ICRVPTR,IXPTR)                                SORTCK2   21
      CALL DMGET(ICOM1+9,ICRVPTR,IYPTR)                                SORTCK2   22
      KOMPCOD = 700000000018 + SHIFT(NPOINTS,18)                       SORTCK2   23
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                 SORTCK2   24
      CALL DMGET(KOMPCOD,IXPTR,X)                                      SORTCK2   25
      CALL DMGET(KOMPCOD,IYPTR,Y)                                      SORTCK2   26
C                                                                      SORTCK2   27
C         CHECK TO SEE IF THE POINTS NEED TO BE SORTED. THEY SHOULD BE SORTCK2   28
C         SORTED WITH THE SORT KEY EQUAL TO THE INDEPENDENT VARIABLE   SORTCK2   29
C         SUBMITTED BY THE USER. IF THEY ARE ALREADY SORTED THAT WAY, NO SORTCK2  30
C         SORT IS NECESSARY. IF NOT, THEY MUST BE SORTED.              SORTCK2   31
C                                                                      SORTCK2   32
      CALL DMGET(ICOM1+10,ICRVPTR,IVAR)                                SORTCK2   33
      IF(IVAR .NE. 0) GO TO 10                                         SORTCK2   34
      IVAR = 1HX                                                       SORTCK2   35
      CALL DMSET(ICOM1+10,ICRVPTR,IVAR)                                SORTCK2   36
   10 IF(IVK .EQ. IVAR) RETURN                                         SORTCK2   37
C                                                                      SORTCK2   38
C         THE POINTS ARE NOT SORTED AS SPECIFIED BY THE USER. SORT     SORTCK2   39
C         THEM.                                                        SORTCK2   40
C                                                                      SORTCK2   41
      CALL SORT2(X,Y,NPOINTS,IVK)                                      SORTCK2   42
      RETURN                                                           SORTCK2   43
      END                                                              SORTCK2   44
```

```
      SUBROUTINE SORT2(X,Y,NPOINTS,IV)                                  SORT2      2
C     MODIFICATION OF SORTER                                            SORT2      3
C                                                                       SORT2      4
C         THIS ROUTINE SORTS THE COORDINATE ARRAYS                      SORT2      5
C         THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT        SORT2      6
C         VARIABLE.                                                     SORT2      7
C                                                                       SORT2      8
C         NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.  SORT2      9
C              IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE SORT2     10
C                       TWO ARRAYS ARE SORTED SO THAT THE             SORT2     11
C                       X - COORDINATE ARRAY IS IN ASCENDING ORDER.   SORT2     12
C                       IF IV = 1HY, THEN THE TWO ARRAYS ARE SORTED SO SORT2     13
C                       THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER. SORT2 14
C                                                                       SORT2     15
      DIMENSION X(1),Y(1)                                               SORT2     16
      DO 10 L=1,NPOINTS,L                                               SORT2     17
      M = 2 * L - 1                                                     SORT2     18
   10 CONTINUE                                                          SORT2     19
C                                                                       SORT2     20
C         START LOOP                                                    SORT2     21
C                                                                       SORT2     22
   20 M = M / 2                                                         SORT2     23
      IF(M .EQ. 0) GO TO 70                                             SORT2     24
      K = NPOINTS - M                                                   SORT2     25
      DO 60 J=1,K                                                       SORT2     26
      L = J                                                             SORT2     27
   30 IF(L .LT. 1) GO TO 60                                             SORT2     28
C                                                                       SORT2     29
C         USE THE SORT KEY.                                             SORT2     30
C                                                                       SORT2     31
      IF(IV .EQ. 1HY) GO TO 40                                          SORT2     32
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORT2     33
      GO TO 50                                                          SORT2     34
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORT2     35
C                                                                       SORT2     36
C         EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.               SORT2     37
C                                                                       SORT2     38
   50 TEMP = X(L)                                                       SORT2     39
      X(L) = X(L+M)                                                     SORT2     40
      X(L+M) = TEMP                                                     SORT2     41
C                                                                       SORT2     42
C         EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.               SORT2     43
C                                                                       SORT2     44
      TEMP = Y(L)                                                       SORT2     45
      Y(L) = Y(L+M)                                                     SORT2     46
      Y(L+M) = TEMP                                                     SORT2     47
C                                                                       SORT2     48
      L = L - M                                                         SORT2     49
      GO TO 30                                                          SORT2     50
   60 CONTINUE                                                          SORT2     51
      GO TO 20                                                          SORT2     52
   70 RETURN                                                            SORT2     53
      END                                                               SORT2     54
```

```
      SUBROUTINE SQUEEZ(NPOINTS,NVALID)                                  SQUEEZ      2
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/    2
      J=0                                                                SQUEEZ      4
      DO 10 I=1,NPOINTS                                                  SQUEEZ      5
      IF(NREC(I).LE.0)GO TO 10                                           SQUEEZ      6
      J=J+1                                                              SQUEEZ      7
      X(J)=X(I)                                                          SQUEEZ      8
      Y(J)=Y(I)                                                          SQUEEZ      9
   10 CONTINUE                                                           SQUEEZ     10
      NVALID=J                                                           SQUEEZ     11
      END                                                                SQUEEZ     12
```

```
1                SUBROUTINE LINK(IBD1,IBONEW,JPREV,JNEXT)                    LINK        2
                 C     ATTACHES LINK IBONEW TO STRING BEGINNING WITH IBD1     LINK        3
                 C     POINTER TO PREVIOUS BEAD IN WORD JPREV                 LINK        4
                 C     POINTER TO NEXT BEAD IN WORD JNEXT                     LINK        5
5                      DATA ICOMP/0100000000C08/                             LINK        6
                       NEXT=IBD1                                             LINK        7
                       IPREV=IBD1                                            LINK        8
                       IF(IBD1*IBONEW.NE. 0)GO TO 10                         LINK        9
                       WRITE (6,5)                                           CYBTTT      24
10                   5 FORMAT(27H NULL BEAD ADDRESSES )                      LINK       11
                       STOP                                                  LINK       12
                    10 CONTINUE                                              LINK       13
                       CALL GMGET(ICOMP+JNEXT,NEXT,NEXTAD)                   LINK       14
                       IF(NEXTAD.EQ.0)GO TO 20                               LINK       15
15                     IPREV=NEXTAD                                          LINK       16
                       NEXT=NEXTAD                                           LINK       17
                       GO TO 10                                              LINK       18
                    20 CALL GMSET(ICOMP+JNEXT,NEXT,IBONEW)                   LINK       19
                       CALL GMSET(ICOMP+JPREV,IBONEW,IPREV)                  LINK       20
20                     END                                                   LINK       21
```

```
1                SUBROUTINE RLSPLT(IBEAD1,IBEAD,IVALU)                          RLSPLT     2
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG       30AUG      1
           C     J=N*2**18  OMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/    3
                 COMMON/POINTS/X(248),Y(248),NREC(248)                          /POINTS/   2
5                EXTERNAL RLSCRV                                                RLSPLT     5
                 CALL CMGET(ICOM1+11,IBEAD,ICRV)                                5OCT     108
                 CALL SPLICE2(ICRV,RLSCRV,2,0,1,2,1)                            RLSPLT    11
                 END                                                            RLSPLT    17
```

```
1                SUBROUTINE RLSCRV(IBEAD1,IBEAD,IVALU)                          RLSCRV     2
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG       30AUG      1
               C J=N*2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/   3
                 COMMON/POINTS/X(248),Y(248),NREC(248)                          /POINTS/   2
5                JJ=12*2**18                                                    RLSCRV     5
                 CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                RLSCRV     6
                 CALL DMRLBD(NREC(7),NREC(8),NREC(9))                           TOM       36
                 IF(NREC(11).NE.0)CALL DMRLBD(NREC(11))                         TOM       37
                 END                                                            RLSCRV     8
```

```
      1                SUBROUTINE SPLICE(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)          SPLICE     2
                     C SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU   SPLICE     3
                     C IN WORD JMATCH OF EACH BEAD.                                    SPLICE     4
                     C WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND           SPLICE     5
      5              C THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                   SPLICE     6
                     C IF(IS.NE.1)BEAD IS NOT SPLICED OUT                              SPLICE     7
                     C POINTER TO PREVIOUS BEAD IN WORD JPREV                          SPLICE     8
                     C POINTER TO NEXT BEAD IN WORD JNEXT                              SPLICE     9
                       DATA ICOMP/2100000000008/                                       SPLICE    10
     10                NEXT=IBD1                                                       SPLICE    11
                       IPREV=IBD1                                                      SPLICE    12
                       IF(IBD1.NE.0)GOTO10                                             SPLICE    13
                       PRINT5                                                          SPLICE    14
                     5 FORMAT(13H IBD1 IS NULL )                                       SPLICE    15
     15                IS=2                                                            SPLICE    16
                       RETURN                                                          SPLICE    17
                    10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                           SPLICE    18
                       CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                             SPLICE    19
                       IF(ICKVALU.EQ.IVALU) GOTO15                                     SPLICE    20
     20                IF(NEXTAD.EQ.0)GOTO12                                           SPLICE    21
                       IPREV=NEXT                                                      SPLICE    22
                       NEXT=NEXTAD                                                     SPLICE    23
                       GO TO 10                                                        SPLICE    24
                    12 WRITE (6,6) IVALU                                               CYB777    21
     25             6  FORMAT(19H NO MATCH FOR IVALU ,O20)                            SPLICE    26
                       IS=2                                                            SPLICE    27
                       RETURN                                                          SPLICE    28
                    15 CONTINUE                                                        SPLICE    29
                     C IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL       SPLICE    30
     30                IF(NEXT.EQ.IBD1)GO TO 25                                        SPLICE    31
                     C OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE   SPLICE    32
                       IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                 SPLICE    33
                       IF(NEXTAD.EQ.0)GOTO22                                           SPLICE    34
                       CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                             SPLICE    35
     35                IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                SPLICE    36
                    22 CONTINUE                                                        SPLICE    37
                       CALL SUBR(IBD1,NEXT,IVALU)                                      SPLICE    38
                       IF(IS.EQ.1)CALL DMRLBD(NEXT)                                    SPLICE    39
                       RETURN                                                          SPLICE    40
     40             25 CALL SUBR(IBD1,IBD1,IVALU)                                     SPLICE    41
                       IF(NEXTAD.EQ.0)GO TO 40                                         SPLICE    42
                       NEXT=NEXTAD                                                     SPLICE    43
                    30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                             SPLICE    44
                       CALL SUBR(IBD1,NEXT,IVALU)                                      SPLICE    45
     45                IF(IS.EQ.1)CALL DMRLBD(NEXT)                                    SPLICE    46
                       IF(NEXTAD.EQ.0)GO TO 40                                         SPLICE    47
                       NEXT=NEXTAD                                                     SPLICE    48
                       GO TO 30                                                        SPLICE    49
                    40 CONTINUE                                                        SPLICE    50
     50                IF(IS.EQ.1)CALL DMRLBD(IBD1)                                    SPLICE    51
                       END                                                             SPLICE    52
```

```
      FUNCTION YVA(NP,NDEG,IFIT,ICODE,IV,IER,XX)                        16AUG76   55
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
C     IFIT, 1 LEAST SQUARES, 2 SPLINE, 3 AITKEN                         26AUG      18
C     ICODE, 1 COORD, 2 SLOPES, 3 MIN, 4 DELTA, 5 MAX, 6 TRIM           10MAY77    62
C     IER GT 1 MEANS BAD INTERP                                         16AUG76   59
C     (IER.EQ.6)MEANS ARGUMENT OUT OF RANGE                             29APR      23
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        29APR      24
C                                                                       19APR      1
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        19APR      2
C                                                                       19APR      3
C     IER=0  SUCCESSFUL INTERPOLATION                                   19APR      4
C     IER=1  SUCCESSFUL EXTRAPOLATION                                   19APR      5
C     IER=2  ARGUMENT OUT OF BOUNDS                                     19APR      6
C     IER=3  ATTEMPTING CURVE EVALUATION WITHOUT FIT                    19APR      7
C     IER=4  ATTEMPTING SLOPES CROSSPLOT WITHOUT SPLINE FIT             19APR      8
      IF (IFIT.NE.2) GO TO 120                                          29APR      25
      IF(IV.EQ.1HY)GO TO 105                                            29APR      26
      IF(XX.LT.X(1))GO TO 110                                           29APR      27
      IF(XX.GT.X(NP))GO TO 110                                          29APR      28
      GO TO 120                                                         29APR      29
  105 IF(XX.LT.Y(1))GO TO 110                                           29APR      30
      IF(XX.GT.Y(NP))GO TO 110                                          29APR      31
      GO TO 120                                                         29APR      32
  110 IER=6                                                             29APR      33
      RETURN                                                            29APR      34
  120 CONTINUE                                                          29APR      35
      XV=-4.                                                            19APR      9
      YV=0.                                                             19APR      10
      IF(IFIT.LT.1)GO TO 60                                             16AUG76    62
      IF(IFIT.GT.3)GO TO 60                                             16AUG76    63
      IER=0                                                             16AUG76    64
      GO TO (2,12,30,2,28,2)ICODE                                       10MAY77    63
    2 CONTINUE                                                          16AUG76    66
      GO TO (10,15,35)IFIT                                              16AUG76    67
   10 YY=P(XX,NREC,NP)                                                  16AUG76    68
      GO TO 50                                                          16AUG76    69
   12 NTYP=2                                                            16AUG76    70
      IF(IFIT.NE.1)GO TO 20                                             19APR      11
      YY=PP(XX,NREC,NP)                                                 19APR      12
      GO TO 50                                                          19APR      13
   15 NTYP=1                                                            16AUG76    72
      GO TO 20                                                          16AUG76    73
   20 CONTINUE                                                          26AUG      19
      IF(IFIT.EQ.2)GO TO 21                                             19APR      14
C     SLOPES CROSSPLOT PERMITTED ONLY WITH SPLINE CURVEFIT              19APR      15
      CALL ERROFM(XV,YV-1.,41HSLOPES CROSSPLOT REQUIRES SPLINE CURVEFIT, 19APR      16
     141,7HYVA    ,6HINFORM)                                            19APR      17
      CALL ABTSKC(7HOTRBACK)                                            14JUNE     72
   21 CONTINUE                                                          19APR      20
      IF(IV.EQ.1HY)GO TO 25                                             26AUG      20
      YY=SPLIN2(XX,NP,X,Y,NREC,NTYP)                                    26AUG      21
      GO TO 50                                                          26AUG      22
   25 YY=SPLIN2(XX,NP,Y,X,NREC,NTYP)                                    26AUG      23
      GO TO 50                                                          16AUG76    75
   28 CONTINUE                                                          10MAY77    64
      IF(IV.NE.1HY)GO TO 29                                             10MAY77    65
      YY=FMAX(Y,NP)                                                     10MAY77    66
```

```
                      GO TO 50                                                          10MAY77    67
                29    YY=AMAX(Y,NP)                                                     10MAY77    68
60                    GO TO 50                                                          10MAY77    69
                31    CONTINUE                                                          26AUG      24
                      IF(IV.NE.1HY)GO TO 32                                             20SEPT     152
                      YY=AMIN(X,NP)                                                     26AUG      26
                      GO TO 50                                                          26AUG      27
65              32    YY=AMIN(Y,NP)                                                     26AUG      28
                      GO TO 50                                                          16AUG76    77
                35    CONTINUE                                                          26AUG      29
                      IF(IV.EQ.1HY)GO TO 37                                             26AUG      30
                      CALL INTERP (X,Y,NP,NDEG,XX,YY,NREC,IER)                          LASTBUX    20
70                    GO TO 50                                                          26AUG      32
                37    CALL INTERP (Y,X,NP,NDEG,XX,YY,NREC,IER)                          LASTBUX    21
                      GO TO 50                                                          16AUG76    79
                50    YVA=YY                                                            16AUG76    80
                      IF(IER.EQ.0)RETURN                                                19APR      21
75                    IF(IER.EQ.1)GO TO 49                                              19APR      22
                      CALL ERROR#(XV,YV-1.,21HARGUMENT OUT OF RANGE  ,21, 3HYVA  ,      19APR      23
                     1 6HINFORM)                                                        19APR      24
                      CALL RETSKC(7HQTRBACK)                                            14JUNE     73
                49    CONTINUE                                                          19APR      26
80                    RETURN                                                            16AUG76    81
                51    IER=3                                                             19APR      29
                      CALL ERROR#(XV,YV-1.,40HATTEMPTING TO EVALUATE CURVE WITHOUT FIT , 19APR      30
                     1 40,3HYVA  ,6HINFORM)                                             19APR      31
                      CALL RETSKC(7HQTRBACK)                                            14JUNE     77
85                    END                                                               16AUG76
```

```
      FUNCTION PP(Z,C,N)                                                    19APR   32
C     COMPUTES DERIVATIVE OF POLYNOMIAL IN Z WITH COEFFICIENTS IN C         19APR   33
C     DEGREE IS N-1                                                         19APR   34
      DIMENSION C(N)                                                        19APR   35
      PP=C(2)                                                               19APR   36
      DO 10 I=3,N                                                           19APR   37
      PP=PP+C(I)*Z**(I-2)                                                   19APR   38
   10 CONTINUE                                                              19APR   39
      RETURN                                                                19APR   40
      END                                                                   19APR   41
```

```
      SUBROUTINE LSQ25(X,XS,YS,T,DT,N,LLL,NOTAB,NPBI,NPEI,IBUG,TOL)      21JUNE76   3
      DIMENSION XS(N),YS(N,1),T(1),DT(1)                                 LSQ25      3
      REAL *KK                                                           LSQ25      4
      NPB=NPBI                                                           21JUNE76   4
      NPE=NPEI                                                           21JUNE76   5
      IF(NPB.GT.N)NPB=N                                                  21JUNE76   6
      IF(NPE.GT.N)NPE=N                                                  21JUNE76   7
  101 LLL=0                                                              LSQ25      5
      T=1                                                                LSQ25      6
      K=N                                                                LSQ25      7
      KKK=F                                                              LSQ25      8
      IF(N.LT.5)KKK=N                                                    LSQ25      9
      IF(XS(1)-XS(N))10,10,20                                            LSQ25     10
   10 A=1.0                                                              LSQ25     11
      GOTO2101                                                           LSQ25     12
   20 A=-1.0                                                             LSQ25     13
 2101 JJ=(I+K)/2                                                         LSQ25     14
      IF(I-JJ)22,25,22                                                   LSQ25     15
   22 IF(A*(X-XS(JJ)))23,24,24                                           LSQ25     16
   23 K=JJ                                                               LSQ25     17
      GOTO2101                                                           LSQ25     18
   24 I=JJ                                                               LSQ25     19
      GOTO2101                                                           LSQ25     20
   25 IF(I-2)26,26,27                                                    LSQ25     21
   26 IMIN=1                                                             LSQ25     22
      IMAX=NPB                                                           LSQ25     23
      GOTO30                                                             LSQ25     24
   27 IF(N-I-2)29,29,29                                                  LSQ25     25
   28 IMIN=N-NPE+1                                                       LSQ25     26
      IMAX=N                                                             LSQ25     27
      GOTO30                                                             LSQ25     28
   29 IMIN=I-2                                                           LSQ25     29
      IMAX=I+2                                                           LSQ25     30
   30 SX=0.0                                                             LSQ25     31
      SX2=0.0                                                            LSQ25     32
      SX3=0.0                                                            LSQ25     33
      SX4=0.0                                                            LSQ25     34
      DO31M=IMIN,IMAX                                                    LSQ25     35
      SX=SX+XS(M)                                                        LSQ25     36
      SX2=SX2+XS(M)**2                                                   LSQ25     37
      SX3=SX3+XS(M)**3                                                   LSQ25     38
   31 SX4=SX4+XS(M)**4                                                   LSQ25     39
      D12=KKK*SX2-SX**2                                                  LSQ25     40
      IF(ABS(D12).GE.TOL)GOTO3111                                        LSQ25     41
      PRINT 103,D12                                                      LSQ25     42
      LLL=LLL+1                                                          LSQ25     43
  103 FORMAT(*  D12= *,E16.8)                                            LSQ25     44
 3111 CONTINUE                                                           LSQ25     45
      D13=KKK*SX3-SX2*SX                                                 LSQ25     46
      D23=SX*SX3-SX2**2                                                  LSQ25     47
      D=SX2*D23-SX3*D13+SX4*D12                                          LSQ25     48
      IF(ABS(D).GE.TOL)GO TO 3112                                        LSQ25     49
      PRINT 104,D                                                        LSQ25     50
      LLL=LLL+2                                                          LSQ25     51
  104 FORMAT(*  D= *,E16.8)                                              LSQ25     52
 3112 CONTINUE                                                           LSQ25     53
      DO32K=1,NOTAB                                                      LSQ25     54
```

```
      SY=0.0                                                              LSQ25   55
      SXY=0.0                                                             LSQ25   56
60    SX2Y=0.0                                                            LSQ25   57
      DO 3109 M=IMIN,IMAX                                                 LSQ25   58
      SY=SY+YS(M,K)                                                       LSQ25   59
      SXY=SXY+YS(M,K)*XS(M)                                               LSQ25   60
      SX2Y=SX2Y+YS(M,K)*XS(M)**2                                          LSQ25   61
65  3109 CONTINUE                                                         LSQ25   62
      DC=SY*D23-SXY*D13+SX2Y*D12                                          LSQ25   63
      C=DC/D                                                              LSQ25   64
      DB=KKK*(SXY-C*SX3)-SX*(SY-C*SX2)                                    LSQ25   65
      B=DB/D12                                                            LSQ25   66
70    A=(SY-B*SX-C*SX2)*0.2                                               LSQ25   67
      IF(IBUG)3101,3110,3101                                              LSQ25   68
 3101 WRITE(6,3102)          I,IMIN,IMAX                                  LSQ25   69
 3102 FORMAT(1X ,5HDEBUG, 3I8  )                                          LSQ25   70
      WRITE(6,3103)          A,B,C,D3,DC,D12,D13,D23,SX,SX2,SX3,SX4,      LSQ25   71
75   1SY,SXY,SX2Y,D                                                      LSQ25   72
 3103 FORMAT(1X,9E14.7)                                                   LSQ25   73
 3110 Y(K)=A+B*X+C*X**2                                                   LSQ25   74
   32 DY(K)=B+2.0*C*X                                                    LSQ25   75
      RETURN                                                              LSQ25   76
80    END                                                                 LSQ25   77
```

```
      SUBROUTINE FITBLK (ICVBD,N,NDEG1,IFIT,IV1)                        FIX2      4
C     IFIT, 1 LSQ, 2 SPLINE, 3 AITKEN                                   16AUG76  17
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/  2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG     1
C     J=M*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/   3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1    10MAY77  61
     1,IDELTF,IDELBD(2)                                                 FLCA02    5
      IFIT=0                                                            16AUG76  20
      CALL DMGET(ICOM1+11,ICVBD,IFITBD)                                 16AUG76  21
      IF(IFITBD.NE.0)GO TO 2                                            50CT     101
      CALL ERROPM(-3.,3.,34HCURVE FITS REQUIRED FOR CROSSPLOTS ,34,     50CT     102
     1'FITBLK',3HINFORM)                                                50CT     103
      CALL ABTSKD(7HOTRBACK)                                            50CT     104
    2 CONTINUE                                                          50CT     105
      CALL DMGET(ICOM1+1,IFITBD,ITYPE)                                  16AUG76  23
      IF(ITYPE.EQ.10HLEASTSQUAR)IFIT=1                                  16AUG76  24
      IF(ITYPE.EQ.6HSPLINE)IFIT=2                                       16AUG76  25
      IF(ITYPE.EQ.6HAITKEN)IFIT=3                                       16AUG76  26
      CALL DMGET(ICOM1+5,IFITBD,N)                                      16AUG76  27
      JJ=M*2**18                                                        16AUG76  28
      CALL DMGET (ICOM1+10,ICVBD,IV1)                                   FIX2      5
      GO TO (10,20,30)IFIT                                              16AUG76  30
   10 CONTINUE                                                          16AUG76  31
      CALL DMGET(ICOM7+JJ+5,IFITBD,NREC)                                16AUG76  32
      IF (JT.NE.3.AND.JT.NE.5) GO TO 11                                 LASTBUX  11
      CALL DMGET (ICOM1+5,ICVBD,N)                                      LASTBUX  12
      JJ=N*2**18                                                        LASTBUX  13
C                                                                       LASTBUX  14
      CALL DMGET (ICOM1+9,ICVBD,IXPTR)                                  LASTBUX  15
      CALL DMGET (ICOM1+9,ICVBD,IYPTR)                                  LASTBUX  16
      CALL DMGET (ICOM7+JJ,IXPTR,X)                                     LASTBUX  17
      CALL DMGET (ICOM7+JJ,IYPTR,Y)                                     LASTBUX  18
   11 CONTINUE                                                          LASTBUX  19
      RETURN                                                            16AUG76  33
   20 CONTINUE                                                          16AUG76  34
      CALL DMGET(ICOM7+JJ+5,IFITBD,NREC)                                16AUG76  35
      KK=JJ+N+5                                                         16AUG76  36
      LL=JJ+2*N+5                                                       16AUG76  37
      CALL DMGET(ICOM7+KK,IFITBD,X)                                     16AUG76  39
      CALL DMGET(ICOM7+LL,IFITBD,Y)                                     16AUG76  40
      RETURN                                                            16AUG76  41
   30 CALL DMGET (ICOM1+6,IFITBD,NDEG1)                                 FIX2      6
      KK=JJ+6                                                           16AUG76  46
      LL=JJ+N+6                                                         16AUG76  47
      CALL DMGET(ICOM7+KK,IFITBD,X)                                     16AUG76  49
      CALL DMGET(ICOM7+LL,IFITBD,Y)                                     16AUG76  50
      RETURN                                                            16AUG76  51
      END                                                               16AUG76  54
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C  -  -  A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE    SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 8 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 8                                          SPLIN2    27
       KK = K                                                            SPLIN2    28
       GO TO 120                                                         SPLIN2    29
    8  CONTINUE                                                          SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN                                                             SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1        1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3   SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
       RETURN                                                            SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
      FUNCTION P(Z,C,N)                                                 P          2
C                                                                       P          3
C         THIS ROUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES P          4
C         POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".          P          5
C                                                                       P          6
C         Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                P          7
C         C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.   P          8
C         N - I - THE NUMBER OF COEFFICIENTS.                           P          9
C                                                                       P         10
      DIMENSION C(N)                                                    P         11
      NDEG = N - 1                                                      P         12
      P=C(1)                                                            18JUNE76  21
      DO 10 I=2,N                                                       18JUNE76  22
      P=P+C(I)*Z**(I-1)                                                 18JUNE76  23
   10 CONTINUE                                                          P         16
      RETURN                                                            P         17
      END                                                               P         18
```

```
1                FUNCTION AMIN(A,N)                                  AMIN      2
                 DIMENSION A(N)                                      AMIN      3
                 V=A(1)                                              AMIN      4
                 DO 10 I=2,N                                         AMIN      5
5                IF(A(I).LT.V)V=A(I)                                 AMIN      6
              10 CONTINUE                                            AMIN      7
                 AMIN=V                                              AMIN      8
                 END                                                 AMIN      9
```

```
1                SUBROUTINE SEEK(X,XTAB,NVAL,NRX)                    SEEK      2
                 DIMENSION XTAB(1)                                   SEEK      3
                 DO 10 I=1,NVAL                                      SEEK      4
                 IF(X.NE.XTAB(I))GO TO 10                            SEEK      5
5                NRX=I                                               SEEK      6
                 RETURN                                              SEEK      7
10               CONTINUE                                            SEEK      8
                 NRX=0                                               SEEK      9
                 END                                                 SEEK     10
```

```
      SUBROUTINE SKALBD(ICELB)                                          3FEB      30
      COMMON/TDIS/AA(4),BB(8)                                           3FEB      31
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      DATA XL/0./,YL/6./                                                3FEB      33
      DDY=.1*(BB(8)-BB(7))                                              3FEB      34
      BB(3)=YMIN=(BB(7)-DDY)                                            3FEB      35
      YMAX=BB(4)+DDY                                                    3FEB      36
      BB(5)=(YMAX-YMIN)/YL                                              3FEB      37
      CALL DMSET(ICOM1+8,IDELB,ISCLBD)                                  3FEB      38
      JJ=8*2**18                                                        3FEB      39
      CALL DMGTBD(8,ISCXBD)                                             15FEB     10
      CALL DMSET(ICOM7+JJ,ISCXBD,BB)                                    15FEB     11
      CALL DMSET (ICOM1+8,IDELB,ISCXBD)                                 15FEB     12
      END                                                               3FEB      44
```

```
      SUBROUTINE RWPLGR(IBEAD1,INEXT,IPREV,IONE)                        RWPLGR     2
      COMMON/BEADS/IBGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      EXTERNAL SUBGET                                                   RWPLGR     5
C     IONE=1 MEANS ONLY ONE PLOT IN GROUP                               RWPLGR     6
      IONE=0                                                            RWPLGR     7
      CALL DMGET(ICOM1+IPREV,IBEAD1,IPR)                                RWPLGR     8
      IF(IPR.NE.0)GO TO 10                                              RWPLGR     9
      CALL DMGET(ICOM1+INEXT,IBEAD1,NEXT)                               RWPLGR    10
      IF(NEXT.NE.0)RETURN                                               RWPLGR    11
      IONE=1                                                            RWPLGR    12
      RETURN                                                            RWPLGR    13
   10 CALL SPLICE(IBEAD1,SUBGET,IPREV,0,IPREV,INEXT,0)                  RWPLGR    14
      END                                                               RWPLGR    15
```

```
1                SUBROUTINE SUBGET(IBEAD1,IBEAD,IVALU)                    RWPLGR     16
                 IBEAD1=IBEAD                                             RWPLGR     17
                 END                                                      RWPLGR     18
```

```
1                SUBROUTINE PARSEL(P1,IBEAD,P2)                                  2NOV      153
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG        30AUG       1
                C J=N*2**12  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/     3
                 COMMON/PXVAL/PPVALU(10),IPPV,PXNAM,NCRVS2                       16NOV      20
5                NCRVS2=NCRVS2+1                                                 16NOV      21
                 CALL DMGET(ICOM1+4,IBEAD,PPVALU(NCRVS2))                        16NOV      22
                 END                                                             2NOV      158
```

```
      1                SUBROUTINE DEL8D(IDEL8)                                  DEL8D      2
                 C     CONSTRUCT DELTA BEAD                                     DEL8D      3
                       COMMON/TOIS/AA(4),CCC(8)                                 20SEPT   128
                       COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1  10MAY77   61
      5               1,IDELTE,IDEL8D(2)                                       FLCAD2     5
                       COMMON/DISKT/I8D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG 30AUG      1
                 C     J=N*2**18  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)  /DISKT/   3
                       COMMON/POINTS/X(248),Y(248),NREC(248)                    /POINTS/   2
                       IF(JT.GT.1)GO TO 5                                       3FEB      42
     10                CALL DMGT8D(14,IDEL8)                                    DEL8D      7
                       JJ=14*2**18                                              DEL8D      8
                       CALL DMGET(ICOM7+JJ,ICPOSS1,NREC)                        DEL8D      9
                       CALL DMGT8D(12,ICRVDEL)                                  DEL8D     10
                       NREC(11)=ICRVDEL                                         DEL8D     12
     15                CALL DMGT8D(10,JTITLE)                                   15FEB     13
                       NREC(10)=IDEL8                                           15FEB     14
                       NREC(6)=JTITLE                                           16NOV     24
                       CALL DMSET(ICOM1+3,JTITLE,10H     DELTA )                29APR     21
                       NREC(1)=0                                                5OCT      74
     20                TAX=NREC(7)                                              5OCT      75
                       CALL DMGT8D(7,I8X)                                       5OCT      76
                       NREC(7)=I8X                                              5OCT      77
                       CALL DMSET(ICOM7+JJ,IDEL8,NREC)                          DEL8D     13
                       CALL LINK(ICROSS1,IDEL8,3,1)                             DEL8D     14
     25                ICRV1=ICRVDEL                                            3FEB      43
                       JK=7*2**18                                               3FEB      44
                       CALL DMGET(ICOM7+JK,IAX,XX)                              3FEB      45
                       XX(7)=8HDELTA OF                                         3FEB      46
                       CALL DMSET(ICOM7+JK,I8X,XX)                              3FEB      47
     30                GO TO 6                                                  3FEB      48
                  5    CONTINUE                                                 3FEB      49
                       CALL DMGT8D(12,ICRVDEL)                                  3FEB      50
                       CALL LINK(ICRV1,ICRVDEL,2,1)                             3FEB      51
                  6    CONTINUE                                                 3FEB      52
     35                JJ=12*2**18                                              20SEPT    129
                       CALL DMGET(ICOM7+JJ,ICEL8D(1),NREC)                      DEL8D     15
                       NREC(1)=0                                                DEL8D     16
                       NREC(2)=0                                                DEL8D     17
                       NREC(6)=JT                                               3FEB      53
     40                NDPTS=NREC(5)                                            DEL8D     19
                       IXX=NREC(8)                                              20SEPT    130
                       IPY1=NREC(9)                                             DEL8D     20
                       CALL DMGT8D(NDPTS,IDELY)                                 DEL8D     21
                       NREC(9)=IDELY                                            15FEB     15
     45                CALL DMGT8D(NDPTS,IDELX)                                 5OCT      78
                       NREC(8)=IDELX                                            5OCT      79
                       IRP=NREC(7)                                              5OCT      80
                       CALL DMGT8D(NDPTS,IRP)                                   5OCT      81
                       NREC(7)=IRP                                              5OCT      82
     50                JJ=NDPTS*2**18                                           20SEPT    131
                       CALL DMGET(ICOM1+9,IDEL8D(2),IPY2)                       DEL8D     25
                       CALL DMGET(ICOM7+JJ,IPY1,Y)                              DEL8D     26
                       CALL DMGET(ICOM7+JJ,IPY2,X)                              DEL8D     27
                       CALL DMGET(ICOM1+4,IDEL8D(1),P1VAL)                      20SEPT    132
     55                CALL DMGET(ICOM1+4,IDEL8D(2),P2VAL)                      20SEPT    133
                       ENCODE(17,100,NREC(3))P1VAL                              3FEB      54
                  100  FORMAT(F10.2)                                            27MAY       6
```

```
                JK=12*2**18                                          15FEB    16
                CALL DMSET(ICOM7+JK,ICRVDEL,NREC)                    15FEB    17
60              CALL DMSET (ICOM1+4,ICRVDEL,P2VAL)                   3FEB     57
                DO 10 I=1,NDPTS                                      DELBD    28
             10 Y(I)=Y(I)-X(I)                                       DELBD    29
                CALL DMSET(ICOM7+JJ,ICELY,Y)                         DELBD    30
                CALL DMGET(ICOM7+JJ,IXX,X)                           20SEPT   138
65              CALL DMGET(ICOM7+JJ,IDR,NREC)                        5OCT     87
                CALL DMSET(ICOM7+JJ,IRP,NREC)                        5OCT     88
                CALL DMSET(ICOM7+JJ,ICELX,X)                         5OCT     89
                C1=AMIN(Y,NDPTS)                                     3FEB     58
                IF(C1.LT.CCC(7))CCC(7)=C1                            3FEB     59
70              C1=AMAX(Y,NDPTS)                                     3FEB     60
                IF(C1.GT.CCC(8))CCC(8)=C1                            3FEB     61
                END                                                  DELBD    31
```

```
      SUBROUTINE DELPIC(IPLTPTR,IDELBD)                                  DELPIC      2
      DIMENSION IDELBD(2)                                                DELPIC      3
      COMMON/PARVAL/VALUES(20),IVBEAD(20),NVAL,NAME                      DELPIC      4
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/    2
      COMMON/DISKT/I9D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG       1
C     J=N*2**18  OMSET(ICOM7+J,I9DA,DATA)  STORES DATA(N) IN I8DA(1)     /DISKT/     3
      EXTERNAL PIKK                                                      DELPIC      7
      DIMENSION IPICK(2)                                                 DELPIC      8
      CALL OMGET(ICOM1+11,IPLTPTR,ICRV)                                  DELPIC      9
      NVAL=0                                                             DELPIC     10
      CALL SPLICE(ICRV,PIKK,2,0,1,2,0)                                   DELPIC     11
      IF(NVAL.GE.2)GO TO 5                                               50CT       96
      CALL ERROFM(1.,-3.,58HAT LEAST TWO PARAMETER VALUES REQUIRED FOR D 50CT       97
     1ELTA CROSSPLOT ,58,6HDELPIC,6HINFORM)                              50CT       98
      CALL ABTSKC (7HOTBBACK)                                            LASTBUX     9
    5 CONTINUE                                                           50CT      100
      CALL PICKY(-5.,5.,43HPICK PARAM1,PARAM2,FOR DELTA(PARAM1-PARAM2),  DELPIC     12
     143,VALUES,NVAL,IPICK,2,NREC,NAME)                                  DELPIC     13
      DO 10 I=1,2                                                        DELPIC     14
   10 IDELBD(I)=IVBEAD(IPICK(I))                                         DELPIC     15
      END                                                                DELPIC     16
```

```
1                SUBROUTINE PIKK(IBXX,IBEAD,IVALU)                                  PIKK      2
                 COMMON/PARVAL/VALUES(20),IVBEAD(20),NVAL,NAME                      PIKK      3
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG     1
           C     J=N*2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/   3
5                NVAL=NVAL+1                                                        PIKK      5
                 IVBEAD(NVAL)=IBEAD                                                 PIKK      6
                 CALL DMGET(ICOM1+4,IBEAD,VALUES(NVAL))                             PIKK      7
                 CALL DMGET(ICOM1+3,IBEAD,NAME)                                     PIKK      8
                 END                                                                PIKK      9
```

```
      SUBROUTINE PICKY(XOR,YOR,MSSG,NC,VALUES,NVAL,IPICK,NPIC,IDUM,NAME)  PICKY      2
      DIMENSION MSSG(1),VALUES(NVAL),IPICK(NPIC),IDUM(1)                  PICKY      3
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                    CYB777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77    3
C     (XOR,YOR) INCHES. LOCATION OF UPPER LEFT CORNER OF DISPLAY-INPUT     PICKY      5
C     MSSG - ALPHA INSTRUCTION MESSAGE WITH NC CHARACTERS - INPUT          PICKY      6
C     VALUES - AN ARRAY CONTAINING NVAL VALUES - INPUT                     PICKY      7
C     IPICK - AN ARRAY RETURNING THE INDICES OF THE NPIC ITEMS PICKED      PICKY      8
C     IDUM - A SCRATCH ARRAY WITH NVAL+3+2*NPIC LOCATIONS                  PICKY      9
      CALL ORIGIN(XOR,YOR)                                                PICKY     10
      YV=0.                                                               PICKY     11
      YOFF=.5                                                             PICKY     12
      CALL ANNOTA(0.,YV,MSSG,NC,IDUM(NVAL+1),IGN,-0,-0,-0)                PICKY     13
      YV=YV-YOFF                                                          PICKY     14
      CALL ANNOTA(0.,YV,NAME,10,IDUM(NVAL+3),IGN,-0,-0,-0)                PICKY     15
      DO 20 I=1,NVAL                                                      PICKY     16
      ENCODE (10,100,IDUMMY) VALUES(I)                                    20SEPT    146
      YV=YV-YOFF                                                          PICKY     18
   20 CALL ANNOTA(0.,YV,IDUMMY,10,IDUM(I),ISTR,I,-0,-0)                   PICKY     19
      CALL ANNOTA(0.,YV-1.,6HACCEPT,6,IDUM(NVAL+2),IBUT,66,-0,-0)         PICKY     20
    7 CALL GIBUT(0,NCON,IDOTR,IDOCR)                                      PICKY     21
      IF(IDOCR.EQ.66)GO TO 8                                              PICKY     22
      GO TO 7                                                             PICKY     23
    8 NR=NVAL                                                             20SEPT    147
      CALL GIFSID(NCON,NR,IDUM(NVAL+4),IBUF)                              20SEPT    148
      IF(NR.EQ.NPIC)GO TO 9                                               PICKY     25
      CALL ERROR(0.,YV-2.,13HREPICK VALUES,13,5HPICKY,6HINFORM)           PICKY     26
      GO TO 7                                                             PICKY     27
    9 CONTINUE                                                            PICKY     28
      DO 30 I=1,NR                                                        PICKY     29
   30 IPICK(I)=IBUF(I)                                                    20SEPT    149
      NERAS=NVAL+3                                                        20SEPT    150
      CALL ERASE(IDUM,NERAS)                                              PICKY     32
  100 FORMAT(G10.3)                                                       PICKY     33
      END                                                                 PICKY     34
```

```
      SUBROUTINE SAVEC(IBEAD1)                                          SAVEC      2
      COMMON/DISKT/IBC1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /CISKT/    3
      COMMON /TTEMP/TEMP(10)                                            /TTEMP/    2
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10) /CROSTB/    2
     1,EMACH,PVALU(10),PNAME(10),IRP(10)                                /CROSTB/    3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1    10MAY77    61
     1,IDELTF,IDELBD(2)                                                 FLCAD2     5
      COMMON/MULTKS/NKS,AXK(20)                                         13SEPT     66
      DO 10 I=1,NCRVS                                                   20SEPT     144
      NPTS=NCVAL(I)                                                     15FEB      18
      IF(NPTS.GT.10)NPTS=10                                             15FEB      19
      JJ=NPTS*2**18                                                     15FEB      20
      CALL DMGTBD(NPTS,IRPTR)                                           15FEB      21
      CALL DMSET(ICOM7+JJ,IRPTR,IRP)                                    SAVEC      8
      IF (NPTS.EQ.0) GO TO 10                                           3AUG       14
      CALL DMGTBD(12,IBEAD)                                             SAVEC      10
      IF(NKS.LE.1)GO TO 11                                              13SEPT     67
      PNAME(I)=THK                                                      13SEPT     68
      PVALU(I)=AXK(I)                                                   13SEPT     69
   11 CONTINUE                                                          13SEPT     70
      CALL DMSET(ICOM1+3,IBEAD,PNAME(I))                                SAVEC      11
      CALL DMSET(ICOM1+4,IBEAD,PVALU(I))                                SAVEC      12
      CALL DMSET(ICOM1+5,IBEAD,NPTS)                                    SAVEC      14
      CALL DMSET(ICOM1+6,IBEAD,ISYM(I))                                 SAVEC      15
      CALL DMSET(ICOM1+7,IBEAD,IRPTR)                                   SAVEC      16
      CALL DMGTBD(NPTS,IMPTR)                                           SAVEC
      CALL DMSET(ICOM1+8,IBEAD,IMPTR)                                   SAVEC
      CALL DMGTBD(NPTS,IYPTR)                                           SAVEC      19
      CALL DMSET(ICOM1+9,IBEAD,IYPTR)                                   SAVEC      20
      JJ=NPTS*2**18                                                     SAVEC      21
      CALL DMSET(ICOM7+JJ,IMPTR,XM(1,I))                                SAVEC      22
      CALL DMSET(ICOM7+JJ,IYPTR,YM(1,I))                                SAVEC      23
      CALL DMSET(ICOM1+10,IBEAD,IVC)                                    20SEPT     145
      IF(IBEAD1.NE.0)GO TO 8                                            SAVEC      24
      IBEAD1=IBEAD                                                      SAVEC      25
      GO TO 10                                                          SAVEC      26
    8 CALL LINK(IBEAD1,IBEAD,2,1)                                       SAVEC      27
   10 CONTINUE                                                          SAVEC      28
      END                                                               SAVEC      29
```

```
      OVERLAY(16,0)                                                      STOPIT      2
      PROGRAM STOPIT                                                     STOPIT      3
C                                                                        STOPIT      4
C                                                                        STOPIT      5
C     THIS ROUTINE CALLS THE CALCOMP PLOTTING SEQUENCE AND FINISHES      STOPIT      6
C     THE GRAPHICS JOB                                                   STOPIT      7
C                                                                        STOPIT      8
      COMMON /GP/ ISUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CY8777      1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                  6JULY77     3
      COMMON /PLOTCOM/ IFIRST,IPLTFL1,NCPC,ICALCD,DPLOT(128)             /PLOTCOM/   2
      CALL DOCUSE                                                        2NOV      159
      IF(IFIRST.NE.0)CALL PLOTE(N)                                       14JUNE     75
      CALL DMFLSH                                                        STOPIT     14
      CALL GICNJB (NCON)                                                 STOPIT     15
      CALL GIBUT(1,NCON,IDOT)                                            STOPIT     16
      CALL GICNSL (NCON)                                                 STOPIT     17
      STOP                                                               STOPIT     18
      END                                                                STOPIT     19
```

```
      SUBROUTINE DOCUSE                                                 DOCUSE      2
      COMMON /GP/ IBUF(129),NCON,NBYTE,MBYTE,IGN,ISIN,                  CY8777      1
     X     ISTR,IOUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77     3
      CPTIM=SECOND(CP)-CPTO                                             6JULY77    28
      DAY=DATE(X)                                                       DOCUSE      4
      TIM=TIME(X)                                                       DOCUSE      5
      IR=1                                                              DOCUSE      8
   40 READ (43,100) ITEXT                                               DOCUSE      9
  100 FORMAT(A10)                                                       DOCUSE     10
      IF (EOF(43)) 60,50                                                DOCUSE     11
   50 IR=IR+1                                                           DOCUSE     12
      GO TO 40                                                          DOCUSE     13
   60 BACKSPACE 43                                                      DOCUSE     14
      CALL PERMFIL (IER,5HALTER,6HTAPE43)                               DOCUSE     15
      WRITE (43,110) IR,DAY,TIM,CPTIM                                   16NOV      25
  110 FORMAT(1X,I5,4X,A10,5X,A10,5X,F10.2)                              DOCUSE     17
      ENDFILE 43                                                        DOCUSE     18
      CALL PERMFIL (IERR,5HALTER,6HTAPE43)                              DOCUSE     19
      RETURN                                                            DOCUSE     20
      END                                                               DOCUSE     21
```

```
1                OVERLAY(26,0)                                          14JUNE     76
                 PROGRAM PLOTC                                          14JUNE     77
                 COMMON /PLOTCOM/ IFIRST,IPLTFL1,NCPC,ICALBD,DPLOT(128) /PLOTCOM/   2
                 IF(IFIRST.EQ.1)GO TO 5                                 14JUNE     78
5                CALL SPLOT(IFIRST,DPLOT)                               14JUNE     79
               5 CALL SUBPFL2 (IBEAU1,ICALBD,IVALU)                     LASTBUG     8
                 CALL RETSKC (*HOTRBACK)                                14JUNE     82
                 END                                                    PLOTC      10
```

```
      1                 SUBROUTINE SPLICE2(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)          SPLICE2     2
                      C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU  SPLICE2     3
                      C     IN WORD JMATCH OF EACH BEAD.                                   SPLICE2     4
                      C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND          SPLICE2     5
      5               C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                 SPLICE2     6
                      C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                             SPLICE2     7
                      C     POINTER TO PREVIOUS BEAD IN WORD JPREV                         SPLICE2     8
                      C     POINTER TO NEXT BEAD IN WORD JNEXT                             SPLICE2     9
                            DATAICOMP/2100000000000B/                                      SPLICE2    10
     10                     NEXT=IBD1                                                      SPLICE2    11
                            IPREV=IBD1                                                     SPLICE2    12
                            IF(IBD1.NE.0)GOTO10                                            SPLICE2    13
                            WRITE (6,5)                                                    CYB777     22
                       5    FORMAT(13H IBD1 IS NULL )                                      SPLICE2    15
     15                     IS=2                                                           SPLICE2    16
                            RETURN                                                         SPLICE2    17
                      10    CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                          SPLICE2    18
                            CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                            SPLICE2    19
                            IF(ICKVALU.EQ.IVALU) GOTO15                                    SPLICE2    20
     20                     IF(NEXTAD.EQ.0)GOTO12                                          SPLICE2    21
                            IPREV=NEXT                                                     SPLICE2    22
                            NEXT=NEXTAD                                                    SPLICE2    23
                            GO TO 10                                                       SPLICE2    24
                         12 WRITE (6,6) IVALU                                              CYB777     23
     25                  6 FORMAT(19H NO MATCH FOR IVALU ,O20)                             SPLICE2    26
                            IS=2                                                           SPLICE2    27
                            RETURN                                                         SPLICE2    28
                      15    CONTINUE                                                       SPLICE2    29
                      C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL      SPLICE2    30
     30                     IF(NEXT.EQ.IBD1)GO TO 25                                       SPLICE2    31
                      C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE  SPLICE2    32
                            IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                SPLICE2    33
                            IF(NEXTAD.EQ.0)GOTO22                                          SPLICE2    34
                            CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                            SPLICE2    35
     35                     IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)               SPLICE2    36
                      22    CONTINUE                                                       SPLICE2    37
                            CALL SUBR(IBD1,NEXT,IVALU)                                     SPLICE2    38
                            IF(IS.EQ.1)CALL DMRLBD(NEXT)                                   SPLICE2    39
                            RETURN                                                         SPLICE2    40
     40                     CALL SUBR(IBD1,IBD1,IVALU)                                     SPLICE2    41
                            IF(NEXTAD.EQ.0)GO TO 40                                        SPLICE2    42
                            NEXT=NEXTAD                                                    SPLICE2    43
                      30    CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                            SPLICE2    44
                            CALL SUBR(IBD1,NEXT,IVALU)                                     SPLICE2    45
     45                     IF(IS.EQ.1)CALL DMRLBD(NEXT)                                   SPLICE2    46
                            IF(NEXTAD.EQ.0)GO TO 40                                        SPLICE2    47
                            NEXT=NEXTAD                                                    SPLICE2    48
                            GO TO 30                                                       SPLICE2    49
                      40    CONTINUE                                                       SPLICE2    50
     50                     IF(IS.EQ.1)CALL DMRLBD(IBD1)                                   SPLICE2    51
                            END                                                            SPLICE2    52
```

```
      FUNCTION P(Z,C,N)                                                 P          2
C                                                                       P          3
C     THIS ROUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES     P          4
C     POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".              P          5
C                                                                       P          6
C     Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                    P          7
C     C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.       P          8
C     N - I - THE NUMBER OF COEFFICIENTS.                               P          9
C                                                                       P         10
      DIMENSION C(N)                                                    P         11
      NDEG = N - 1                                                      P         12
      P=C(1)                                                            18 JUNE 76 21
      DO 10 I=2,N                                                       18 JUNE 76 22
      P=P+C(I)*Z**(I-1)                                                 18 JUNE 76 23
   10 CONTINUE                                                          P         16
      RETURN                                                            P         17
      END                                                               P         18
```

```
      FUNCTION YVA(NP,NDEG,IFIT,ICODE,IV,IER,XX)                        16AUG76   55
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
C     IFIT, 1 LEAST SQUARES, 2 SPLINE, 3 AITKEN                         26AUG     18
C     ICODE, 1 COORD, 2 SLOPES, 3 MIN, 4 DELTA, 5 MAX, 6 TRIM           10MAY77   62
C     IER GT 1 MEANS BAD INTERP                                         16AUG76   59
C     (IER.EQ.6)MEANS ARGUMENT OUT OF RANGE                             29APR     23
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        29APR     24
C                                                                       19APR      1
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        19APR      2
C                                                                       19APR      3
C     IER=0  SUCCESSFUL INTERPOLATION                                   19APR      4
C     IER=1  SUCCESSFUL EXTRAPOLATION                                   19APR      5
C     IER=2  ARGUMENT OUT OF BOUNDS                                     19APR      6
C     IER=3  ATTEMPTING CURVE EVALUATION WITHOUT FIT                    19APR      7
C     IER=4  ATTEMPTING SLOPES CROSSPLOT WITHOUT SPLINE FIT             19APR      8
      IF (IFIT.NE.2) GO TO 120                                          29APR     25
      IF(IV.EQ.1HY)GO TO 105                                            29APR     26
      IF(XX.LT.X(1))GO TO 110                                           29APR     27
      IF(XX.GT.X(NP))GO TO 110                                          29APR     28
      GO TO 120                                                         29APR     29
  105 IF(XX.LT.Y(1))GO TO 110                                           29APR     30
      IF(XX.GT.Y(NP))GO TO 110                                          29APR     31
      GO TO 120                                                         29APR     32
  110 IER=6                                                             29APR     33
      RETURN                                                            29APR     34
  120 CONTINUE                                                          29APR     35
      XX=-4.                                                            19APR
      YY=0.                                                             19APR
      IF(IFIT.LT.1)GO TO 60                                             16AUG76   62
      IF(IFIT.GT.3)GO TO 60                                             16AUG76   63
      IER=0                                                             16AUG76   64
      GO TO (2,12,30,2,29,2)ICODE                                       10MAY77   65
    2 CONTINUE                                                          16AUG76   66
      GO TO (10,15,35)IFIT                                              16AUG76   67
   10 YY=P(XX,NREC,NP)                                                  16AUG76   68
      GO TO 50                                                          16AUG76   69
   12 NTYP=2                                                            16AUG76   70
      IF(IFIT.NE.1)GO TO 20                                             19APR     11
      YY=PP(XX,NREC,NP)                                                 19APR     12
      GO TO 50                                                          19APR     13
   15 NTYP=1                                                            16AUG76   72
      GO TO 20                                                          16AUG76   73
   20 CONTINUE                                                          26AUG     19
      IF(IFIT.EQ.2)GO TO 21                                             19APR     14
C     SLOPES CROSSPLOT PERMITTED ONLY WITH SPLINE CURVEFIT              19APR     15
      CALL ERROFM(XV,YV-1.,41HSLOPES CROSSPLOT REQUIRES SPLINE CURVEFIT, 19APR     16
     141,3HYVA     ,6HINFORM)                                           19APR     17
      CALL AETSKC(7HQTRBACK)                                            14JUNE    72
   21 CONTINUE                                                          19APR     20
      IF(IV.EQ.1HY)GO TO 25                                             26AUG     20
      YY=SPLIN2(XX,NP,X,Y,NREC,NTYP)                                    26AUG     21
      GO TO 50                                                          26AUG     22
   25 YY=SPLIN2(XX,NP,Y,X,NREC,NTYP)                                    26AUG     23
      GO TO 50                                                          16AUG76   75
   28 CONTINUE                                                          10MAY77   64
      IF(IV.NE.1HY)GO TO 29                                             10MAY77   65
      YY=AMAX(Y,NP)                                                     10MAY77   66
```

```
                GO TO 50                                                          10MAY77   67
          29    YY=AMAX(Y,NP)                                                     10MAY77   68
60              GO TO 50                                                          10MAY77   69
          30    CONTINUE                                                          26AUG     24
                IF(IV.NE.1HY)GO TO 32                                             20SEPT    152
                YY=AMIN(X,NP)                                                     26AUG     26
                GO TO 50                                                          26AUG     27
65        32    YY=AMIN(Y,NP)                                                     26AUG     28
                GO TO 50                                                          16AUG76   77
          35    CONTINUE                                                          26AUG     29
                IF(IV.EQ.1HY)GO TO 37                                             26AUG     30
                CALL INTERP (X,Y,NP,NDEG,XX,YY,NREC,IER)                          LASTBUX   20
70              GO TO 50                                                          26AUG     32
          37    CALL INTERP (Y,X,NP,NDEG,XX,YY,NREC,IER)                          LASTBUX   21
                GO TO 50                                                          16AUG76   79
          50    YVA=YY                                                            16AUG76   80
                IF(IER.EQ.0)RETURN                                                19APR     21
75              IF(IER.EQ.1)GO TO 49                                              19APR     22
                CALL ERROR4(XV,YV-1.,21HARGUMENT OUT OF RANGE  ,21, 3HYVA  ,      19APR     23
               1 6HINFORM)                                                        19APR     24
                CALL GETSKC(7HOTRBACK)                                            14JUNE    73
          49    CONTINUE                                                          19APR     26
80              RETURN                                                            16AUG76   81
          60    IER=3                                                             19APR     29
                CALL ERROR4(XV,YV-1.,40HATTEMPTING TO EVALUATE CURVE WITHOUT FIT , 19APR     30
               1 3HYVA  ,6HINFORM)                                                19APR     31
                CALL GETSKC(7HOTRBACK)                                            14JUNE    74
85              END                                                               16AUG76   83
```

```
1                FUNCTION PP(Z,C,N)                                          19APR    32
                 C  COMPUTES DERIVATIVE OF POLYNOMIAL IN Z WITH COEFFICIENTS IN C  19APR    33
                 C  DEGREE IS N-1                                             19APR    34
                    DIMENSION C(N)                                            19APR    35
5                   PP=C(2)                                                   19APR    36
                    DO 10 I=3,N                                               19APR    37
                    PP=PP+C(I)*Z**(I-2)                                       19APR    38
                 10 CONTINUE                                                  19APR    39
                    RETURN                                                    19APR    40
10                  END                                                       19APR    41
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C  -  -  A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE    SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 8 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 8                                          SPLIN2    27
       KK = K                                                            SPLIN2    28
       GO TO 120                                                         SPLIN2    29
    8 CONTINUE                                                           SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN    0                                                        SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1        1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3   SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
      RETURN                                                             SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
1                FUNCTION AMIN(A,N)                                    AMIN       2
                 DIMENSION A(N)                                        AMIN       3
                 V=A(1)                                                AMIN       4
                 DO 10 I=2,N                                           AMIN       5
5                IF(A(1).LT.V)V=A(I)                                   AMIN       6
            10   CONTINUE                                              AMIN       7
                 AMIN=V                                                AMIN       8
                 END                                                   AMIN       9
```

```
1                FUNCTION AMAX(A,N)                                    AMAX      2
                 DIMENSION A(N)                                        AMAX      3
                 V=A(1)                                                AMAX      4
                 DO 10 I=2,N                                           AMAX      5
5                IF(A(I).GT.V)V=A(I)                                   AMAX      6
              10 CONTINUE                                              AMAX      7
                 AMAX=V                                                AMAX      8
                 END                                                   AMAX      9
```

```
      SUBROUTINE SUBCPL2(IBEAD1,IBEAD,IVALU)                            SUBCPL2    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      COMMON /TTEMP/TEMP(10)                                            /TTEMP/    2
      COMMON/SYMT/XXX,YYY,DELYYY,HGHT                                   SUBCPL2    6
      COMMON/NCCC/NC                                                    20SEPT   154
      DIMENSION BTITL(4),TITL(4)                                        SUBCPL2    7
      EXTERNAL SUBCOR2                                                  SUBCPL2    8
      DATA BTITL /1H ,1H ,1H ,1H /                                      SUBCPL2    9
      DO 10 I=1,4                                                       SUBCPL2   10
   10 TITL(I)=BTITL(I)                                                  SUBCPL2   11
      CALL DMGET(ICOM1+6,IBEAD,ITITL)                                   SUBCPL2   12
      XXX=9.                                                            SUBCPL2   13
      YYY=6.                                                            SUBCPL2   14
      DELYYY=.21                                                        SUBCPL2   15
      HGHT=.105                                                         SUBCPL2   16
      IF(ITITL.EQ.0)GO TO 20                                            SUBCPL2   17
      CALL DMGET(ICOM1+2,ITITL,NC)                                      SUBCPL2   18
      NW=NC/10                                                          SUBCPL2   19
      IF((NC-10*NW).GT.0)NW=NW+1                                        SUBCPL2   20
      JJ=NW*2**18+2                                                     SUBCPL2   21
      CALL DMGET(ICOM7+JJ,ITITL,TEMP)                                   SUBCPL2   22
      IF(NW.GT.4)NW=4                                                   SUBCPL2   23
      DO 11 I=1,NW                                                      SUBCPL2   24
   11 TITL(I)=TEMP(I)                                                   SUBCPL2   25
   20 CONTINUE                                                          SUBCPL2
      CALL DMGET(ICOM1+7,IBEAD,IAXIS)                                   SUBCPL2
      CALL DMGET(ICOM1+5,IAXIS,XLABL)                                   SUBCPL2   28
      CALL DMGET(ICOM1+7,IAXIS,YLABL)                                   SUBCPL2   29
      CALL DMGET(ICOM1+8,IBEAD,IAXISC)                                  SUBCPL2   30
      JJ=2*2**18                                                        SUBCPL2   31
      CALL DMGET(ICOM7+JJ,IAXISC,TEMP)                                  SUBCPL2   32
      CALL SETPLT(XLABL,YLABL,TITL,TEMP(1),TEMP(6),TEMP(3),TEMP(8),     SUBCPL2   33
     1 9.,6.,TEMP(2),TEMP(4))                                           SUBCPL2   34
      CALL FRCOND (IBEAD)                                               15FEB      22
      CALL DMGET(ICOM1+11,IPEAD,ICRVPTR)                                SUBCPL2   35
      IF(ICRVPTR.EQ.0)RETURN                                            SUBCPL2   36
      CALL SPLICE2(ICRVPTR,SUBCOR2,2,0,1,2,0)                           SUBCPL2   37
      END                                                               SUBCPL2   38
```

525

```
      SUBROUTINE SUBCCR2(IBEAD1,IBEAD,IVALU)                            SUBCCR2    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**15  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NPEC(248)                             /POINTS/   2
      COMMON/SYMT/XXX,YYY,DELYYY,HGHT                                   SUBCCR2    5
      COMMON /PLTPAS/ XMN,DX,YMN,DY,XLEN                                15FEB     23
      CALL DMGET (ICOM1+12,IBEAD,ISTAT)                                 3FEB      62
      IF (ISTAT.EQ.9HINACTIVE) RETURN                                   3FEB      63
      CALL DMGET(ICOM1+3,IBEAD,NAME)                                    SUBCCR2    6
      CALL DMGET(ICOM1+4,IBEAD,VALUE)                                   SUBCCR2    7
      CALL DMGET(ICOM1+5,IBEAD,N)                                       SUBCCR2    8
      JJ=N*2**15                                                        SUBCCR2    9
      CALL DMGET (ICOM1+7,IBEAD,INR)                                    FRISK      1
      CALL DMGET (ICOM7+JJ,INR,NREC)                                    FRISK      2
      CALL DMGET(ICOM1+8,IBEAD,IXP)                                     SUBCCR2   10
      CALL DMGET(ICOM7+JJ,IXP,X)                                        SUBCCR2   11
      CALL DMGET(ICOM1+9,IBEAD,IYP)                                     SUBCCR2   12
      CALL DMGET(ICOM7+JJ,IYP,Y)                                        SUBCCR2   13
      CALL DMGET(ICOM1+6,IBEAD,ISYM)                                    SUBCCR2   14
      CALL DMGET(ICOM1+1,IBEAD,IFIN)                                    SUBCCR2   15
      CALL DMGET(ICOM1+11,IBEAD,ICF80)                                  SUBCCR2   16
      IF(IFIN.EQ.0)IFIN=1                                               SUBCCR2   22
      CALL SYMBOL (XXX+.3,YYY,HGHT,ISYM,0.,-1)                          15FEB     24
      CALL SYMBOL(999.,YYY,HGHT,NAME,0.,10)                             SUBCCR2   24
      CALL NUMBER(999.,YYY,HGHT,VALUE,0.,2)                             SUBCCR2   25
      YYY=YYY-DELYYY                                                    SUBCCR2   26
      CALL LOP (N)                                                      FRISK      3
      CALL PLTCRV(X,Y,N,ISYM,-1,IFIN)                                   SUBCCR2   27
      IF (ICF80.EQ.0) GO TO 10                                          15FEB     25
      CALL DMGET (ICOM1+10,IBEAD,INDVAR)                                15FEB     26
C     CALCOMP OF CURVE FIT                                              15FEB     27
      CALL LCFIT (ICF80,INDVAR)                                         15FEB     28
   10 CONTINUE                                                          15FEB     29
      IF (IFIN.EQ.1) CALL PLOT (XLEN+3.5,0.,-3)                         15FEB     30
      END                                                               SUBCCR2   29
```

526

```
      SUBROUTINE LOF (NPOINTS)                                          LOF     2
C                                                                       LOF     3
C       THIS ROUTINE ELIMINATES FROM THE HARDCOPY THE POINTS THAT HAVE  LOF     4
C       BEEN DELETED OR ARE OUTSIDE THE RANGE OF THE AXES               LOF     5
C                                                                       LOF     6
      COMMON /TTEMP/TEMP(10)                                            /TTEMP/ 2
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/ 2
      EQUIVALENCE (XMIN,TEMP(1)),(XDEL,TEMP(2)),(YMIN,TEMP(3)),         LOF     9
     1    (YDEL,TEMP(4))                                                LOF    10
C                                                                       LOF    11
C                                                                       LOF    12
      XMAX=XMIN+9.*XDEL                                                 LOF    13
      YMAX=YMIN+6.*YDEL                                                 LOF    14
      NPX=0                                                             LOF    15
      DO 50 J=1,NPOINTS                                                 LOF    16
      IF (X(J).LT.XMIN.OR.X(J).GT.XMAX) GO TO 50                        LOF    17
      IF (Y(J).LT.YMIN.OR.Y(J).GT.YMAX) GO TO 50                        LOF    18
      IF (NREC(J).LT.0) GO TO 50                                        LOF    19
      NPX=NPX+1                                                         LOF    20
      X(NPX)=X(J)                                                       LOF    21
      Y(NPX)=Y(J)                                                       LOF    22
   50 CONTINUE                                                          LOF    23
      NPOINTS=NPX                                                       LOF    24
      RETURN                                                            LOF    25
      END                                                               LOF    26
```

```
1                SUBROUTINE SPLOT(IFIRST,DPLOT)                          SPLOT      2
                 C  CALCOMP INITIALIZATION                               SPLOT      3
                 CALL PLOTS (DPLOT,128,99)                               INHALE    89
                 CALL PLOT (0.,-1.,-3)                                   SPLOT      5
5                CALL PLOT(0.,1.,-3)                                     SPLOT      6
                 IFIRST=1                                                SPLOT      7
                 END                                                     SPLOT      8
```

```
      SUBROUTINE SETPLT(XLABL,YLABL,TITL,XMIN,XMAX,YMIN,YMAX                SETPLT      2
     1,XLEN,YLEN,DELX,DELY)                                                 SETPLT      3
C     CALCOMP VERSION                                                       SETPLT      4
C      CALCOMP PLOT INITIATION                                              SETPLT      5
C     MULTIPLE CURVES ARE PERMITTED                                         SETPLT      6
C                                                                           SETPLT      7
C                                                                           SETPLT      8
C     TO PLOT, A CALL IS MADE TO  SETPLOT FOLLOWED BY A SEPARATE CALL       SETPLT      9
C      TO PLTCRV FOR EACH CURVE PLOTTED.                                    SETPLT     10
C      (A SET OF 3 CURVES SHARING THE SAME AXES WOULD REQUIRE A CALL        SETPLT     11
C      TO SETPLT FOLLOWED BY 3 CALLS TO PLTCRV.  DATA FOR ALL THREE         SETPLT     12
C     CURVES MUST LIE IN THE RANGE XMIN TO XMAX AND YMIN TO YMAX)           SETPLT     13
C                                                                           SETPLT     14
C     IF DESIRED, XMIN,XMAX,YMIN AND YMAX MAY BE COMPUTED BY                SETPLT     15
C      CONSECUTIVE CALLS TO MXMNA                                           SETPLT     16
C                                                                           SETPLT     17
C     IF DELX OR DELY ARE ENTERED AS ZERO, SETPLT WILL COMPUTE              SETPLT     18
C     REASONABLE VALUES FOR THEM.                                           SETPLT     19
C                                                                           SETPLT     20
C     IT IS NECESSARY TO MAKE A CALL TO SPLOT BEFORE PLOTTING               SETPLT     21
C      AND A CALL TO EPLOT WHEN FINISHED WITH PLOTTING                      SETPLT     22
C                                                                           SETPLT     23
C     CALCOMP VERSION                                                       SETPLT     24
C                                                                           SETPLT     25
C     TITLE HAS 40 CHARACTERS                                               SETPLT     26
C     AXIS LABELS HAVE 10 CHARACTERS                                        SETPLT     27
C     ERRORS:   STOP 13  IF FLAGGED                                         SETPLT     28
C      X OR Y OUT OF SPECIFIED RANGE (THIS ERROR NOT FLAGGED)               SETPLT     29
C      XLEN.GT.12  - LENGTH OF X-AXIS TOO LONG                              SETPLT     30
C      YLEN.GT.9.  -  Y-AXIS TOO LONG                                       SETPLT     31
C      XMAX .GT. DELX*XLEN + XMIN  -  FATAL DELX ERROR                      SETPLT     32
C      YMAX .GT. DELY*YLEN + YMIN  -  FATAL DELY ERROR                      SETPLT     33
      DIMENSION A(4)                                                        SETPLT     34
      COMMON/PLTPAR/XMN,DX,YMN,DY,XL                                        SETPLT     35
      COMMON/NCCC/NC                                                        20SEPT    160
      DATA XLMAX,YLMAX/12.,9./                                              SETPLT     36
      IF(XLEN.LT.1..OR.XLEN.GT.XLMAX)CALL ERRMSG(10HXLEN ERROR,1)           SETPLT     37
      IF(YLEN.LT.0..OR.YLEN.GT.YLMAX)CALL ERRMSG(10HYLEN ERROR,1)           SETPLT     38
      XMN=XMIN                                                              SETPLT     39
      YMN=YMIN                                                              SETPLT     40
      XL=XLEN                                                               SETPLT     41
      IF(DELX.NE.0.)GO TO 10                                                SETPLT     42
      A(1)=YMIN                                                             SETPLT     43
      A(2)=XMAX                                                             SETPLT     44
      CALL SCALE(A,XL,2,1)                                                  SETPLT     45
      DY=A(4)                                                               SETPLT     46
      GO TO 15                                                              SETPLT     47
   10 DX=DELX                                                               SETPLT     48
      IF(XMAX.GT.DX*XLEN+XMIN)CALL ERRMSG(10HDELX ERROR,1)                  SETPLT     49
   15 IF(DELY.NE.0.)GO TO 20                                                SETPLT     50
      A(1)=YMIN                                                             SETPLT     51
      A(2)=YMAX                                                             SETPLT     52
      CALL SCALE(A,YLEN,2,1)                                                SETPLT     53
      DY=A(4)                                                               SETPLT     54
      GO TO 25                                                              SETPLT     55
   20 DY=DELY                                                               SETPLT     56
      IF(YMAX.GT.DY*YLEN+YMIN)CALL ERRMSG(10HDELY ERROR,1)                  SETPLT     57
```

```
      25  CONTINUE                                                          SETPLT     58
          IF (TITL.NE.1H*) CALL SYMBOL(.2,-.75,.14,TITL,0.,NC)              15FEB      31
60        CALL AXIS(0.,0.,XLABL,-10,XLEN,0.,XMIN,DX)                        SETPLT     60
          CALL AXIS(0.,0.,YLABL,10,YLEN,90.,YMIN,DY)                        SETPLT     61
          END                                                               SETPLT     62
```

530

```
      SUBROUTINE PLTCRV(X,Y,N,ISYM,JMODE,IFIN)                          PLTCRV     2
C                                                                       PLTCRV     3
C     CALCOMP VERSION                                                   PLTCRV     4
C     JMODE  -  PLOT MODE DESIGNATOR, J                                 PLTCRV     5
C       J .LT. 0, DRAW A SYMBOL AT EVERY JTH POINT                      PLTCRV     6
C                                                                       PLTCRV     7
C       J .EQ. 0 - CONNECT ALL POINTS WITH STRAIGHT LINE                PLTCRV     8
C                                                                       PLTCRV     9
C       J .GT. 0, CONNECT ALL POINTS WITH STRAIGHT LINE AND             PLTCRV    10
C                 DRAW A SYMBOL AT EVERY JTH POINT                      PLTCRV    11
C                                                                       PLTCRV    12
C     ISYM - SYMBOL NO. - SEE CALCOMP PLOTTER GUIDE, APPENDIX A         PLTCRV    13
C                                                                       PLTCRV    14
C     ISYM. GT. 13 OR NEGATIVE  -  FATAL SYMBOL NUMBER ERROR            PLTCRV    15
C                                                                       PLTCRV    16
C     IFIN.EQ.1 MEANS LAST CURVE FOR CURRENT AXES                       PLTCRV    17
C                                                                       PLTCRV    18
      DIMENSION X(1),Y(1)                                               PLTCRV    19
      COMMON/PLTPAR/XMN,DX,YMN,DY,XLEN                                  PLTCRV    20
      IF(ISYM.LT.0.OR.ISYM.GT.13)CALL ERRMSG(10HISYM ERROR,1)           PLTCRV    21
C                                                                       14MARCH   23
C        CHECK TO MAKE SURE POINTS ON PLOT                              14MARCH   24
C                                                                       14MARCH   25
      KOUNT=0                                                           14MARCH   26
      XMAX=XMN+6.*DX                                                    29APR     36
      YMAX=YMN+6.*DY                                                    29APR     37
      DO 50 J=1,N                                                       14MARCH
      IF (X(J).GT.XMAX.OR.X(J).LT.XMN) GO TO 50                         29APR
      IF (Y(J).GT.YMAX.OR.Y(J).LT.YMN) GO TO 50                         29APR     39
      KOUNT=KOUNT+1                                                     14MARCH   30
      X(KOUNT)=X(J)                                                     14MARCH   31
      Y(KOUNT)=Y(J)                                                     14MARCH   32
   50 CONTINUE                                                          14MARCH   33
      N=KOUNT                                                           14MARCH   34
      X(N+1)=XMN                                                        PLTCRV    22
      X(N+2)=DX                                                         PLTCRV    23
      Y(N+1)=YMN                                                        PLTCRV    24
      Y(N+2)=DY                                                         PLTCRV    25
      CALL LINE(X,Y,N,1,JMODE,ISYM)                                     PLTCRV    26
      RETURN                                                            15FEB     32
      END                                                               PLTCRV    30
```

```
      SUBROUTINE PRCOND (IPLTPTR)                                          PRCOND      2
C                                                                          PRCOND      3
C     THIS ROUTINE PRINTS THE CONDITIONS AT THE TOP OF THE PLOT            PRCOND      4
C                                                                          PRCOND      5
      COMMON/PGBEAD/IPGNEXT,IPGPREV,IPGTITL,IPGAXIS,IPGCTXT,IPGNCNS,       /PGBEAD/     2
     X              IPGLCA                                                /PGBEAD/     3
      COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,         /PLTBEAD/     2
     X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,        /PLTBEAD/     3
     X               IPFMN,IPFMV                                         /PLTBEAD/     4
      COMMON/COMPCOD/IWDZERO,IWDONE,IWDTWO,IWDTHRE,IWDFOUR,IWDFIVE,        /COMPCOD/     2
     X              IWDSIX,IWDSEVE,IWDEIGH,IWDNINE,IWDTEN                 /COMPCOD/     3
      COMMON/PMBEAD/IPMNEXT,IPMPREV,IPMPLST,IPMMLST,IPMPLOT,IPMPTXT,       /PMBEAD/     2
     X              IPMMTXT                                               /PMBEAD/     3
      DIMENSION ITXT(8)                                                    PRCOND     10
C                                                                          PRCOND     11
C        POSITION AT TOP OF PLOT                                           PRCOND     12
C                                                                          PRCOND     13
      XX=.2                                                                PRCOND     14
      YY=9.                                                                PRCOND     15
      HT=.105                                                              PRCOND     16
C                                                                          PRCOND     17
C        FIND AND DISPLAY CONDITIONS                                       PRCOND     18
      CALL DMGET (IPPG,IPLTPTR,IPGPTR)                                     PRCOND     19
      CALL DMGET (IPGNCNS,IPGPTR,NCOND)                                    PRCOND     20
      IF (NCOND.EQ.0) GO TO 50                                             PRCOND     21
      CALL DMGET (IPGCTXT,IPGPTR,ICTPTR)                                   PRCOND     22
      CALL SYMBOL (XX,YY,HT,12HCONDITION(S),0.,12)                         PRCOND     23
      DO 20 J=1,NCOND                                                      PRCOND     24
      CALL DMGET (IWDTWO,ICTPTR,NCHAR)                                     PRCOND     25
      CALL DMGET(7001000000038,ICTPTR,ITXT)                                PRCOND     26
      YY=YY-.2                                                             PRCOND     27
      IF (YY.LE.6.2) RETURN                                                PRCOND     28
      CALL SYMBOL (XX,YY,HT,ITXT,0.,NCHAR)                                 PRCOND     29
      CALL DMGET (IWDONE,ICTPTR,ICTPTR)                                    PRCOND     30
   20 CONTINUE                                                             PRCOND     31
C                                                                          PRCOND     32
C        MULTIPLIERS                                                       PRCOND     33
   50 CALL DMGET (IPNMULT,IPLTPTR,NMULT)                                   PRCOND     34
      IF (NMULT.EQ.0) RETURN                                               PRCOND     35
      YY=YY-.2                                                             PRCOND     36
      IF (YY.LE.6.2) RETURN                                                PRCOND     37
      CALL SYMBOL (XX,YY,HT,13HMULTIPLIER(S),0.,13)                        PRCOND     38
C                                                                          PRCOND     39
      DO 75 J=1,NMULT                                                      PRCOND     40
      CALL DMGET (IPFMN+2*J-2,IPLTPTR,NAME)                                PRCOND     41
      CALL DMGET (IPFMV+2*J-2,IPLTPTR,VAL)                                 PRCOND     42
      YY=YY-.2                                                             PRCOND     43
      IF (YY.LE.6.2) RETURN                                                PRCOND     44
      CALL SYMBOL (XX,YY,HT,NAME,0.,7)                                     PRCOND     45
      CALL SYMBOL(999.,999.,HT,1H=,0.,1)                                   PRCOND     46
      CALL NUMBER (999.,999.,HT,VAL,0.,2)                                  PRCOND     47
   75 CONTINUE                                                             PRCOND     48
      RETURN                                                               PRCOND     49
      END                                                                  PRCOND     50
```

```
1                SUBROUTINE ERRMSG(MSSG,NW)                          ERRMSG     2
                 DIMENSION MSSG (1)                                  ERRMSG     3
                 WRITE (6,10) (MSSG(I),I=1,NW)                       CY8777    27
              10 FORMAT(1HC,10A10)                                   ERRMSG     5
5                RETURN                                              1SEPT     23
                 END                                                 ERRMSG     7
```

```
      SUBROUTINE CCFIT (ICFPTR,INV)                                      CCFIT      2
C     CALCOMP OF CURVEFIT                                                CCFIT      3
      COMMON/DISKT/I8D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)     /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/   2
      DIMENSION TEMP(4)                                                  15FEB     33
      DIMENSION XF(248),YF(248)                                          CCFIT      7
      JJ=4*2**18                                                         CCFIT      8
      CALL DMGET(ICOM7+JJ,ICFPTR,TEMP)                                   CCFIT      9
      IF(TEMP(1).EQ.1CHLEASTSQUAR) JF=1                                  CCFIT     10
      IF(TEMP(1).EQ.6HSPLINE) JF=2                                       CCFIT     11
      IF(TEMP(1).EQ.6HAITKEN) JF=3                                       CCFIT     12
      CALL DMGET(ICOM1+5,ICFPTR,NP)                                      CCFIT     13
      XN=TEMP(3)                                                         CCFIT     14
      DELX=TEMP(4)                                                       CCFIT     15
      XF(1)=TEMP(2)                                                      CCFIT     16
      JJ=NP*2**18+5                                                      CCFIT     17
      GO TO (10,20,30) JF                                                CCFIT     18
   10 CALL DMGET(ICOM7+JJ,ICFPTR,NREC)                                   CCFIT     19
      GO TO 40                                                           CCFIT     20
   20 CALL DMGET(ICOM7+JJ,ICFPTR,NREC)                                   CCFIT     21
      JJ=JJ+NP                                                           CCFIT     22
      CALL DMGET(ICOM7+JJ,ICFPTR,X)                                      CCFIT     23
      JJ=JJ+NP                                                           CCFIT     24
      CALL DMGET(ICOM7+JJ,ICFPTR,Y)                                      CCFIT     25
      GO TO 40                                                           CCFIT     26
   30 CALL DMGET(ICOM1+6,ICFPTR,NDEG)                                    CCFIT     27
      JJ=JJ+1                                                            CCFIT
      CALL DMGET(ICOM7+JJ,ICFPTR,X)                                      CCFIT     29
      JJ=JJ+NP                                                           CCFIT     30
      CALL DMGET(ICOM7+JJ,ICFPTR,Y)                                      CCFIT     31
   40 CONTINUE                                                           CCFIT     32
      I=1                                                                CCFIT     33
   41 YF(I)=YVA(NP,NDEG,JF,1,INV,IER,XF(I))                              CCFIT     34
      IF(IER.GT.1)YF(I)=0.                                               CCFIT     35
      XF(I+1)=XF(I)+DELX                                                 CCFIT     36
      IF(XF(I+1).GT.XN)GO TO 50                                          CCFIT     37
      I=I+1                                                              CCFIT     38
      GO TO 41                                                           CCFIT     39
   50 NFP=I                                                              CCFIT     40
      IF (INV.EQ.1HX) CALL PLTCRV (XF,YF,NFP,1,0,0)                      20SEPT    162
      IF (INV.EQ.1HY)  CALL PLTCRV (YF,XF,NFP,1,0,0)                     20SEPT    163
      END                                                                CCFIT     42
```

```
      OVERLAY(17,0)                                                          CROSS2     2
      PROGRAM CROSS2                                                         CROSS2     3
      COMMON /GP/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                       CY8777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                      6JULY77    3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1         10MAY77   61
     1,IDELTF,IDELBO(2)                                                     FLCAD2     5
      DIMENSION IDA(10)                                                      CROSS2     8
      COMMON/PXVAL/FPVALU(10),IPPV,PXNAM,NCRVS2                              CROSS2     9
      EXTERNAL PARSEL                                                        CROSS2    10
      COMMON/PLOTCOM/IFIRST,IPLTFL1,NCPC,ICALBO(100)                         CROSS2    11
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)      /CROSTB/   2
     1,FMACH,PVALU(10),PNAME(10),IRP(10)                                     /CROSTB/   3
      COMMON /MULTKS/ MKS,AXK(20)                                            /MULTKS/   2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                     /ITRIMT/   2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR      /BEADS/    2
      EXTERNAL SUBCPLT,RLSPM                                                 CROSS2    17
      EXTERNAL SUBC2                                                         CROSS2    18
      COMMON/TDIS/ITRIM1,PRNAME,NX,NY,XMIN,DELX,YMIN,DELY,XDATMIN,           T3CAUG    21
     1XDATMAX,YDATMIN,YDATMAX                                                T3CAUG    22
      COMMON/DISKT/I3D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               3CAUG      1
C     J=N*2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /CISKT/    3
      DATA XL/9./,YL/6./,BIG/1.0E38/                                         CROSS2    23
      IDELBL(1)=IDELBO(2)=0                                                  CROSS2    27
      ICROSAV=ICROSS1                                                        CROSS2    28
   20 CONTINUE                                                               CROSS2    61
      XMIN=BIG                                                               CROSS2    62
      XMAX=-BIG                                                              CROSS2    63
      YMIN=BIG                                                               CROSS2    64
      YMAX=-BIG                                                              CROSS2    65
      NCRVS=0                                                                CROSS2    66
      DO 15 I=1,10                                                           CROSS2    67
   15   NCVAL(I)=0                                                          CROSS2    68
      CALL ITRIPR(IPLTPTR)                                                   CROSS2    69
 1603 CONTINUE                                                               CROSS2    70
      CALL DMGET(ICOM1+11,IPLTPTR,ICRV1)                                     CROSS2    71
      CALL DMGET(ICOM1+3,ICRV1,PRNAME)                                       CROSS2    72
      CALL DMGET(ICOM1+1,IPLTPTR,IBDNXT)                                     CROSS2    73
      CALL DMGET(ICOM1+7,IPLTPTR,IAXIS)                                      CROSS2    74
      CALL DMGET(ICOM1+5,IAXIS,FMACH)                                        CROSS2    75
      CALL DMGET(ICOM1+7,IAXIS,YNAME)                                        CROSS2    76
      CALL SPLICE(ICRV1,SUBC2,3,PRNAME,1,2,0)                                29APR     40
      DO 300 J=1,NCRVS                                                       CROSS2    80
      K=NCVAL(J)                                                             CROSS2    81
      XMN=AMIN(XM(1,J),K)                                                    CROSS2    82
      IF(XMN.LT.XMIN)XMIN=XMN                                                CROSS2    83
      XMX=AMAX(XM(1,J),K)                                                    CROSS2    84
      IF(XMX.GT.XMAX)XMAX=XMX                                                CROSS2    85
      YMN=AMIN(YM(1,J),K)                                                    CROSS2    86
      IF(YMN.LT.YMIN)YMIN=YMN                                                CROSS2    87
      YMX=AMAX(YM(1,J),K)                                                    CROSS2    88
      IF(YMX.GT.YMAX)YMAX=YMX                                                CROSS2    89
  300 CONTINUE                                                               CROSS2    95
      XDATMIN=XMIN                                                           CROSS2    96
      XDATMAX=XMAX                                                           CROSS2    97
      YDATMIN=YMIN                                                           CROSS2    98
      YDATMAX=YMAX                                                           CROSS2    99
      DDY=.1*(YMAX-YMIN)                                                     CROSS2   100
```

```
                DDY=.1*(YMAX-YMIN)                                      CROSS2   101
                XMIN=XMIN-DDX                                           CROSS2   102
   60           YMIN=YMIN-DDY                                           CROSS2   103
                XMAX=XMAX+DDX                                           CROSS2   104
                YMAX=YMAX+DDY                                           CROSS2   105
                DELX=(XMAX-XMIN)/XL                                     CROSS2   106
                DELY=(YMAX-YMIN)/YL                                     CROSS2   107
   65           CALL CROSP02                                            CROSS2   110
                IF (JT.NE.4) GO TO 500                                  CROSS2   111
                JT=0                                                    CROSS2   112
                YDATMIN=BIG                                             CROSS2   113
                YDATMAX=-BIG                                            CROSS2   114
   70       400 CONTINUE                                                CROSS2   115
                CALL ORIGIN(-5.,5.)                                     CROSS2   116
             402 CALL ANNOTA (0.,0.,26HDELTA CROSS PLOT SELECTION,26,    CROSS2   117
               1     ILA1,IBUT,7,-0,-0)                                 CROSS2   118
                 CALL ANNOTA(0.,-1.,8HCONTINUE,8,IDA2,IBUT,8,-0,-0)     CROSS2   119
   75       405  CALL GIBUT(0,NCON,IDDT,IDDC)                           CROSS2   120
                 CALL GIFAS(IDA1,IDA2)                                  CROSS2   121
                 IF(IDDC.EQ.8)GO TO 500                                 CROSS2   122
                 IF(IDDC.NE.7)GO TO 402                                 CROSS2   123
                 CALL DELPIC(ICROSS1,IDEL8D)                            CROSS2   124
   80            JT=JT+1                                                CROSS2   125
                 CALL DEL8D(IDEL8)                                      CROSS2   126
                 CALL SKAL8D(IDEL8)                                     CROSS2   127
                 GO TO 402                                              CROSS2   128
             500 CONTINUE                                               CROSS2   129
   85            IF(IPDNXT.NE.0)CALL LINK(ICROSS1,IPDNXT,3,1)           27MAY      9
                 CALL DMSET(ICOM1+1,IPLTPTR,ICROSS1)                    27MAY     10
               C DISPLAY CROSSPLOTS                                     CROSS2   132
               C                                                        CROSS2   133
               C SAVE IPLTPTR                                           CROSS2   134
   90            IPPSAVE=IPLTPTR                                        CROSS2   135
                 IPLTPTR=ICROSS1                                        CROSS2   136
                 ICROSS1=ICROSAV                                        CROSS2   137
                 CALL RETSKC(7HFRSTPL2)                                 10MAY77   79
                 END                                                    CROSS2   139
```

```
      SUBROUTINE CROS802                                                    CROSB02    2
      COMMON/DISKT/I8D1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG              30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)        /DISKT/    3
      COMMON/CROST8L/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1        10MAY77   61
     1,IDELTF,IDEL8C(2)                                                     FLGAD2     5
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)     /CROSTB/   2
     1,FMACH,PVALU(10),PNAME(10),IRP(10)                                    /CROSTB/   3
      COMMON /TTEMP/TEMP(10)                                                /TTEMP/    2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                 /POINTS/   2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR     /BEADS/    2
      COMMON/TDIS/ITRIM1,PRNAME,NX,NY,AA(8)                                 CROSB02   12
      COMMON /MULTKS/ NKS,AXK(20)                                           /MULTKS/   2
      CALL DMGTBD(14,ICROSS1)                                               CROSB02   14
      JJ=14*2**18                                                           CROSB02   15
      CALL DMGET(ICOM7+JJ,IPLTPTR,NREC)                                     CROSB02   16
      CALL DMGTBD(7,IPG)                                                    CROSB02   17
      NREC(1)=5                                                             CROSB02   18
      NREC(2)=6                                                             CROSB02   19
      NREC(3)=IPLTPTR                                                       CROSB02   20
      NREC(5)=IPG                                                           CROSB02   21
      CALL DMGTBD(10,ITITL)                                                 CROSB02   22
      NREC(6)=ITITL                                                         CROSB02   23
      CALL DMGTBD(7,IAXIS)                                                  CROSB02   24
      NREC(7)=IAXIS                                                         CROSB02   25
      CALL DMGTBD(8,IAXISC)                                                 CROSB02   26
      NREC(8)=IAXISC                                                        CROSB02   27
      LL=8*2**18                                                            CROSB02   28
      CALL DMSET (ICOM7+LL,IAXISC,AA)                                       CROSB02   29
      CALL DMGET(ICOM1+9,IPLTPTR,IPM)                                       CROSB02   30
      NREC(9)=IPM                                                           CROSB02   31
      NREC(10)=ICROSS1                                                      CROSB02   32
      FREC=FMACH                                                            19APR     45
      IF(IVC.EQ.1HY)FREC=YNAME                                              19APR     46
      CALL DMSET(ICOM7+JJ,ICROSS1,NREC)                                     CROSB02   35
      ICRV1=0                                                               CROSB02   37
      CALL SAVEC2(ICRV1)                                                    CROSB02   38
      CALL DMSET(ICOM1+11,ICROSS1,ICRV1)                                    CROSB02   39
      JJ=7*2**15                                                            CROSB02   40
      CALL DMGET(ICOM1+5,IPLTPTR,IPGOLD)                                    CROSB02   41
      CALL DMGET(ICOM7+JJ,IPGOLD,TEMP)                                      CROSB02   42
      CALL DMSET(ICOM7+JJ,IPG,TEMP)                                         CROSB02   43
      CALL DMSET(ICOM1+3,IPG,ITITL)                                         CROSB02   44
      CALL DMSET(ICOM1+4,IPG,IAXIS)                                         CROSB02   45
      CALL DMSET(ICOM1+4,IAXIS,IPM)                                         CROSB02   46
      CALL DMSET(ICOM1+5,IAXIS,PRNAME)                                      CROSB02   47
      CALL DMSET(ICOM1+6,IAXIS,1)                                           CROSB02   48
      YYNAME=YNAME                                                          CROSB02   49
      IF(IVC.EQ.1HY)YYNAME=FMACH                                            CROSB02   50
      CALL DMSET(ICOM1+7,IAXIS,YYNAME)                                      CROSB02   51
      CALL DMSET(ICOM1+2,ITITL,20)                                          CROSB02   52
      JTT=JT                                                                19APR     47
      IF(JT.EQ.4)JTT=1                                                      19APR     48
      CALL DMSET(ICOM1+3,ITITL,CTYPE(JTT))                                  19APR     49
      CALL DMSET(ICOM1+4,ITITL ,10H CROSSPLOT)                              CROSB02   54
      END                                                                   CROSB02   55
```

```
      SUBROUTINE SAVEC2(IBEAD1)                                          SAVEC2      2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG       1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/     3
      COMMON /TTEMP/TEMP(10)                                             /TTEMP/     2
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)  /CROSTB/     2
     1,FMACH,PVALU(10),PNAME(10),IRP(10)                                 /CROSTB/     3
      COMMON/CROSTBL/ICROSS1,ICRSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1     10MAY77     61
     1,IDELTF,TDELBD(2)                                                  FLCAD2      5
      COMMON /MULTKS/ NKS,AXK(20)                                        /MULTKS/     2
      DO 10 I=1,NCRVS                                                    SAVEC2      11
      NPTS=NCVAL(I)                                                      SAVEC2      12
      IF(NPTS.GT.10)NPTS=10                                              SAVEC2      13
      JJ=NPTS*2**18                                                      SAVEC2      14
      CALL DMGTBD(NPTS,IRPTR)                                            SAVEC2      15
      CALL DMSET(ICOM7+JJ,IRPTR,IRP)                                     SAVEC2      16
      IF (NPTS.EQ.0) GO TO 10                                            SAVEC2      17
      CALL DMGTBD(12,IBEAD)                                              SAVEC2      18
      PNAME(I)=FMACH                                                     19APR       50
      IF(IVC.EQ.1HY)PNAME(I)=YNAME                                       19APR       51
      PVALU(I)=AXK(I)                                                    SAVEC2      21
   11 CONTINUE                                                           SAVEC2      22
      IF(JT.EQ.3)PNAME(I)=7HMINIMUM                                      FIXMIN      10
      IF(JT.EQ.5)PNAME(I)=7HMAXIMUM                                      FIXMIN      11
      CALL DMSET(ICOM1+3,IBEAD,PNAME(I))                                 SAVEC2      23
      IF(JT.EQ.3.OR.JT.EQ.5)GO TO 12                                     FIXMIN      12
      CALL DMSET(ICOM1+4,IBEAD,PVALU(I))                                 SAVEC2      24
   12 CONTINUE                                                           FIXMIN      13
      CALL DMSET(ICOM1+5,IBEAD,NPTS)                                     SAVEC2      25
      CALL DMSET(ICOM1+6,IBEAD,ISYM(I))                                  SAVEC2      26
      CALL DMSET(ICOM1+7,IBEAD,IRPTR)                                    SAVEC2      27
      CALL DMGTBD(NPTS,IMPTR)                                            SAVEC2      28
      CALL DMSET(ICOM1+8,IBEAD,IMPTR)                                    SAVEC2      29
      CALL DMGTBD(NPTS,IYPTR)                                            SAVEC2      30
      CALL DMSET(ICOM1+9,IBEAD,IYPTR)                                    SAVEC2      31
      JJ=NPTS*2**18                                                      SAVEC2      32
      CALL DMSET(ICOM7+JJ,IMPTR,XM(1,I))                                 SAVEC2      33
      CALL DMSET(ICOM7+JJ,IYPTR,YM(1,I))                                 SAVEC2      34
      IF(IBEAD1.NE.0)GO TO 8                                             SAVEC2      35
      IBEAD1=IBEAD                                                       SAVEC2      36
      GO TO 10                                                           SAVEC2      37
    8 CALL LINK(IBEAD1,IBEAD,2,1)                                        SAVEC2      38
   10 CONTINUE                                                           SAVEC2      39
      END                                                                SAVEC2      40
```

```
      SUBROUTINE SUBC2(IBEAD1,IBEAD,IVALU)                                  SUBC2      2
      COMMON/CROSTB/NCRVS,NCVAL(10),IVC,XK,XM(10,10),YM(10,10),ISYM(10)     /CROSTB/    2
     1,FMACH,PVALU(10),PNAME(10),IRP(10)                                    /CROSTB/    3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1        10MAY77    61
     1,IDELTF,IDFLBC(2)                                                     FLOAD2     5
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG              30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                 /POINTS/   2
      COMMON /MULTKS/ NKS,AXK(20)                                           /MULTKS/   2
      CALL DMGET(ICOM1+12,IBEAD,ISTAT)                                      SUBC2     11
      IF(ISTAT.EQ.6HINACTIVE)RETURN                                         SUBC2     12
      CALL FITBLK(IBEAD,NP,NDEG,IFIT,IDVAR)                                 SUBC2     13
      IVC=IDVAR                                                             SUBC2     14
      NCRVS=NKS                                                             SUBC2     15
      DO 18 J=1,NKS                                                         SUBC2     16
      XK=AXK(J)                                                             SUBC2     17
      PVALU(J)=XK                                                           SUBC2     18
      IF(IVC.EQ.1HX)PNAME(J)=FMACH                                          SUBC2     19
      IF(IVC.EQ.1HY)PNAME(J)=YNAME                                          SUBC2     20
      CALL DMGET(ICOM1+4,IBEAD,XXXX)                                        SUBC2     21
      ISYM(J)=J                                                             SUBC2     22
      YV=YVAL(NP,NDEG,IFIT,JT,IDVAR,IER,XK)                                 SUBC2     23
      NCVAL(J)=NCVAL(J)+1                                                   SUBC2     24
      I=NCVAL(J)                                                            SUBC2     25
      IRP(I)=1                                                              SUBC2     26
      XM(I,J)=XXXX                                                          SUBC2     27
      YM(I,J)=1.0                                                           29APR     43
      IF(I.GE.1)YM(I,J)=YV                                                  SUBC2     29
   18 CONTINUE                                                              SUBC2     30
      RETURN                                                                SUBC2     31
      END                                                                   SUBC2     32
```

```
      SUBROUTINE DELBD(IDELB)                                          DELBD     2
C     COMSTRUCT DELTA BEAD                                             DELBD     3
      COMMON/TOLS/AA(4),CCC(8)                                         20SEPT   128
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1   10MAY77   61
     1,IDELTF,ICELBC(2)                                                FLCAD2    5
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG         30AUG     1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)   /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                            /POINTS/   2
      IF(JT.GT.1)GO TO 5                                               3FEB      42
      CALL DMGTBD(14,IDELB)                                            DELBD     7
      JJ=14*2**18                                                      DELBD     8
      CALL DMGET(ICOM7+JJ,ICROSS1,NREC)                                DELBD     9
      CALL DMGTBD(12,ICRVDEL)                                          DELBD    10
      NREC(11)=ICRVDEL                                                 DELBD    12
      CALL DMGTBD(10,JTITLE)                                           15FEB    13
      NREC(10)=IDELB                                                   15FEB    14
      NREC(6)=JTITLE                                                   16NOV    24
      CALL DMSET(ICOM1+3,JTITLE,10H    DELTA )                         29APR    21
      NREC(1)=0                                                        5OCT     74
      IBX=NREC(7)                                                      5OCT     75
      CALL DMGTBD(7,IBX)                                               5OCT     76
      NREC(7)=IBX                                                      5OCT     77
      CALL DMSET(ICOM7+JJ,IDELB,NREC)                                  DELBD    13
      CALL LINK(ICROSS1,IDELB,3,1)                                     DELBD    14
      ICRV1=ICRVDEL                                                    3FEB     43
      JK=7*2**18                                                       3FEB     44
      CALL DMGET(ICOM7+JK,IAX,XX)                                      3FEB
      XX(7)=8HDELTA CP                                                 3FEB
      CALL DMSET(ICOM7+JK,IBX,XX)                                      3FEB     47
      GO TO 6                                                          3FEB     48
    5 CONTINUE                                                         3FEB     49
      CALL DMGTBD(12,ICRVDEL)                                          3FEB     50
      CALL LINK(ICRV1,ICRVDEL,2,1)                                     3FEB     51
    6 CONTINUE                                                         3FEB     52
      JJ=12*2**18                                                      20SEPT   129
      CALL DMGET(ICOM7+JJ,ICELBC(1),NREC)                              DELBD    15
      NREC(1)=0                                                        DELBD    16
      NREC(2)=0                                                        DELBD    17
      NREC(6)=JT                                                       3FEB     53
      NOPTS=NREC(5)                                                    DELBD    19
      IXX=NREC(8)                                                      20SEPT   130
      IPY1=NREC(9)                                                     DELBD    20
      CALL DMGTBD(NOPTS,IDELY)                                         DELBD    21
      NREC(9)=IDELY                                                    15FEB    15
      CALL DMGTBD(NOPTS,IDELX)                                         5OCT     78
      NREC(8)=IDELX                                                    5OCT     79
      IDF=NREC(7)                                                      5OCT     80
      CALL DMGTBD(NOPTS,IRP)                                           5OCT     81
      NREC(7)=IRP                                                      5OCT     82
      JJ=NOPTS*2**18                                                   20SEPT   131
      CALL DMGET(ICOM1+9,IDELBD(2),IPY2)                               DELBD    25
      CALL DMGET(ICOM7+JJ,IPY1,Y)                                      DELBD    26
      CALL DMGET(ICOM7+JJ,IPY2,X)                                      DELBD    27
      CALL DMGET(ICOM1+4,IDELBD(1),P1VAL)                              20SEPT   132
      CALL DMGET(ICOM1+4,IDELBD(2),P2VAL)                              20SEPT   133
      ENCODE(10,100,NREC(3))P1VAL                                      3FEB     54
  100 FORMAT(F10.2)                                                    27MAY     6
```

```
      JK=12*2**18                                                    15FEB    16
      CALL DMSET(ICOM7+JK,ICRVDEL,NREC)                              15FEB    17
60    CALL DMSET (ICOM1+4,ICRVDEL,P2VAL)                             3FE3     57
      DO 10 I=1,NDPTS                                                DELBD    28
   10 Y(I)=Y(I)-X(I)                                                 DELBD    29
      CALL DMSET(ICOM7+JJ,IDELY,Y)                                   DELBD    30
      CALL DMGET(ICOM7+JJ,IXX,X)                                     20SEPT   138
65    CALL DMGET(ICOM7+JJ,IOR,NREC)                                  5OCT     87
      CALL DMSET(ICOM7+JJ,IRP,NREC)                                  5OCT     88
      CALL DMSET(ICOM7+JJ,IDELX,X)                                   5OCT     89
      C1=AMIN(Y,NDPTS)                                               3FEB     58
      IF(C1.LT.CCC(7))CCC(7)=C1                                      3FEB     59
70    C1=AMAX(Y,NDPTS)                                               3FEB     60
      IF(C1.GT.CCC(8))CCC(8)=C1                                      3FEB     61
      END                                                            DELBD    31
```

541

```
      SUBROUTINE SPLICE2(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)          SPLICE2     2
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU     SPLICE2     3
C     IN WORD JMATCH OF EACH BEAD.                                       SPLICE2     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND              SPLICE2     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                     SPLICE2     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                 SPLICE2     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                             SPLICE2     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                 SPLICE2     9
      DATA(ICMP/01000000000000B/                                         SPLICE2    10
      NEXT=IBD1                                                          SPLICE2    11
      IPREV=IBD1                                                         SPLICE2    12
      IF(IBD1.NE.0)GOTO10                                                SPLICE2    13
      WRITE (6,5)                                                        CY8777     22
    5 FORMAT(13H IBD1 IS NULL )                                          SPLICE2    15
      IS=2                                                               SPLICE2    16
      RETURN                                                             SPLICE2    17
   10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                              SPLICE2    18
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                SPLICE2    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                        SPLICE2    20
      IF(NEXTAD.EQ.0)GOTO12                                              SPLICE2    21
      IPREV=NEXT                                                         SPLICE2    22
      NEXT=NEXTAD                                                        SPLICE2    23
      GO TO 10                                                           SPLICE2    24
   12 WRITE (6,6) IVALU                                                  CY8777     23
    6 FORMAT(19H NO MATCH FOR IVALU ,O20)                                SPLICE2    26
      IS=2                                                               SPLICE2    27
      RETURN                                                             SPLICE2    28
   15 CONTINUE                                                           SPLICE2    29
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL          SPLICE2    30
      IF(NEXT.EQ.IBD1)GO TO 25                                           SPLICE2    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE      SPLICE2    32
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                    SPLICE2    33
      IF(NEXTAD.EQ.0)GOTO22                                              SPLICE2    34
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                SPLICE2    35
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                   SPLICE2    36
   22 CONTINUE                                                           SPLICE2    37
      CALL SUBR(IBD1,NEXT,IVALU)                                         SPLICE2    38
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    39
      RETURN                                                             SPLICE2    40
   25 CALL SUBR(IBD1,IBD1,IVALU)                                         SPLICE2    41
      IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    42
      NEXT=NEXTAD                                                        SPLICE2    43
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                SPLICE2    44
      CALL SUBR(IBD1,NEXT,IVALU)                                         SPLICE2    45
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    46
      IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    47
      NEXT=NEXTAD                                                        SPLICE2    48
      GO TO 30                                                           SPLICE2    49
   40 CONTINUE                                                           SPLICE2    50
      IF(IS.EQ.1)CALL DMRLBD(IBD1)                                       SPLICE2    51
      END                                                                SPLICE2    52
```

```
1                FUNCTION AMAX(A,N)                                  AMAX      2
                 DIMENSION A(N)                                      AMAX      3
                 V=A(1)                                              AMAX      4
                 DO 10 I=2,N                                         AMAX      5
5                IF(A(I).GT.V)V=A(I)                                 AMAX      6
              10 CONTINUE                                            AMAX      7
                 AMAX=V                                              AMAX      8
                 END                                                 AMAX      9
```

```
      SUBROUTINE SORT2(X,Y,NPOINTS,IV)                                  SORT2    2
C     MODIFICATION OF SORTER                                            SORT2    3
C                                                                       SORT2    4
C         THIS ROUTINE SORTS THE COORDINATE ARRAYS                      SORT2    5
C         THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT        SORT2    6
C         VARIABLE.                                                     SORT2    7
C                                                                       SORT2    8
C         NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.  SORT2    9
C              IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE SORT2   10
C                       TWO ARRAYS ARE SORTED SO THAT THE             SORT2   11
C                       X - COORDINATE ARRAY IS IN ASCENDING ORDER.   SORT2   12
C                       IF IV = 1HY, THEN THE TWO ARRAYS ARE SORTED SO SORT2   13
C                       THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER. SORT2 14
C                                                                       SORT2   15
      DIMENSION X(1),Y(1)                                               SORT2   16
      DO 10 L=1,NPOINTS,L                                               SORT2   17
      M = 2 * L - 1                                                     SORT2   18
   10 CONTINUE                                                          SORT2   19
C                                                                       SORT2   20
C         START LOOP                                                    SORT2   21
C                                                                       SORT2   22
   20 M = M / 2                                                         SORT2   23
      IF(M .EQ. 0) GO TO 70                                             SORT2   24
      K = NPOINTS - M                                                   SORT2   25
      DO 60 J=1,K                                                       SORT2   26
      L = J                                                             SORT2   27
   30 IF(L .LT. 1) GO TO 60                                             SORT2   28
C                                                                       SORT2   29
C         USE THE SORT KEY.                                             SORT2   30
C                                                                       SORT2   31
      IF(IV .EQ. 1HY) GO TO 40                                          SORT2   32
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORT2   33
      GO TO 50                                                          SORT2   34
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORT2   35
C                                                                       SORT2   36
C         EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.               SORT2   37
C                                                                       SORT2   38
   50 TEMP = X(L)                                                       SORT2   39
      X(L) = X(L+M)                                                     SORT2   40
      X(L+M) = TEMP                                                     SORT2   41
C                                                                       SORT2   42
C         EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.               SORT2   43
C                                                                       SORT2   44
      TEMP = Y(L)                                                       SORT2   45
      Y(L) = Y(L+M)                                                     SORT2   46
      Y(L+M) = TEMP                                                     SORT2   47
C                                                                       SORT2   48
      L = L - M                                                         SORT2   49
      GO TO 30                                                          SORT2   50
   60 CONTINUE                                                          SORT2   51
      GO TO 20                                                          SORT2   52
   70 RETURN                                                            SORT2   53
      END                                                               SORT2   54
```

```
1                SUBROUTINE LINK(I8D1,I8ONEW,JPREV,JNEXT)                LINK      2
           C     ATTACHES LINK I8ONEW TO STRING BEGINNING WITH I8D1      LINK      3
           C     POINTER TO PREVIOUS BEAD IN WORD JPREV                  LINK      4
           C     POINTER TO NEXT BEAD IN WORD JNEXT                      LINK      5
5                DATA ICCMP/O1000000000003/                              LINK      6
                 NEXT=I8D1                                               LINK      7
                 IPREV=I8D1                                              LINK      8
                 IF(I8D1*I8ONEW.NE. 0)GO TO 10                           LINK      9
                 WRITE (6,5)                                             CY8777   24
10               5  FORMAT(21H NULL BEAD ADDRESSES )                     LINK     11
                 STOP                                                    LINK     12
                 10 CONTINUE                                             LINK     13
                 CALL UMGET(ICCMP+JNEXT,NEXT,NEXTAD)                     LINK     14
                 IF(NEXTAD.EQ.0)GO TO 20                                 LINK     15
15               IPREV=NEXTAD                                            LINK     16
                 NEXT=NEXTAD                                             LINK     17
                 GO TO 10                                                LINK     18
                 20 CALL UMSET(ICCMP+JNEXT,NEXT,I8ONEW)                  LINK     19
                 CALL UMSET(ICCMP+JPREV,I8ONEW,IPREV)                    LINK     20
20               END                                                     LINK     21
```

```
1                SUBROUTINE RLSPLT(IBEAD1,IBEAD,IVALU)                          RLSPLT      2
                 COMMON /DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG      30AUG       1
                 J=N*2**19  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/      3
               C COMMON/POINTS/X(248),Y(248),NREC(248)                          /POINTS/    2
5                EXTERNAL RLSCRV                                                RLSPLT      5
                 CALL DMGET(ICOM1+11,IBEAD,ICRV)                                50CT      108
                 CALL SPLICE2(ICRV,RLSCRV,2,0,1,2,1)                            RLSPLT     11
                 END                                                            RLSPLT     17
```

```
1                SUBROUTINE RLSCRV(IBEAD1,IBEAD,IVALU)                          RLSCRV     2
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG       30AUG      1
               C J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1) /DISKT/    3
                 COMMON/POINTS/X(248),Y(248),NREC(248)                          /POINTS/   2
5                JJ=12*2**18                                                    RLSCRV     5
                 CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                RLSCRV     6
                 CALL DMRLBD(NREC(7),NREC(8),NREC(9))                           TOM       36
                 IF(NREC(11).NE.0)CALL DMRLBD(NREC(11))                         TOM       37
                 END                                                            RLSCRV     8
```

```
      SUBROUTINE SPLICE(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)                SPLICE     2
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU          SPLICE     3
C     IN WORD JMATCH OF EACH BEAD.                                            SPLICE     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND                   SPLICE     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                          SPLICE     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                      SPLICE     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                                  SPLICE     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                      SPLICE     9
      DATAICOMP/01000000000008/                                               SPLICE    10
      NEXT=IBD1                                                               SPLICE    11
      IPREV=IBD1                                                              SPLICE    12
      IF(IBD1.NE.0)GOTO10                                                     SPLICE    13
      PRINT5                                                                  SPLICE    14
    5 FORMAT(13H IBD1 IS NULL )                                               SPLICE    15
      IS=2                                                                    SPLICE    16
      RETURN                                                                  SPLICE    17
   10 CALL OMGET(ICOMP+JMATCH,NEXT,ICKVALU)                                   SPLICE    18
      CALL OMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                     SPLICE    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                             SPLICE    20
      IF(NEXTAD.EQ.0)GOTO12                                                   SPLICE    21
      IPREV=NEXT                                                              SPLICE    22
      NEXT=NEXTAD                                                             SPLICE    23
      GO TO 10                                                                SPLICE    24
   12 WRITE (6,6) IVALU                                                       CY8777    21
    6 FORMAT(19H NO MATCH FOR IVALU ,020)                                     SPLICE    26
      IS=2                                                                    SPLICE    27
      RETURN                                                                  SPLICE    28
   15 CONTINUE                                                                SPLICE    29
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL               SPLICE    30
      IF(NEXT.EQ.IBD1)GO TO 25                                                SPLICE    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE           SPLICE    32
      IF(IS.EQ.1)CALL OMSET(ICOMP+JNEXT,IPREV,NEXTAD)                         SPLICE    33
      IF(NEXTAD.EQ.0)GOTO22                                                   SPLICE    34
      CALL OMGET(ICOMP+JPREV,NEXT,IPREAD)                                     SPLICE    35
      IF(IS.EQ.1)CALL OMSET(ICOMP+JPREV,NEXTAD,IPREAD)                        SPLICE    36
   22 CONTINUE                                                                SPLICE    37
      CALL SUBR(IBD1,NEXT,IVALU)                                              SPLICE    38
      IF(IS.EQ.1)CALL OMRLBD(NEXT)                                            SPLICE    39
      RETURN                                                                  SPLICE    40
   25 CALL SUBR(IBD1,IBD1,IVALU)                                              SPLICE    41
      IF(NEXTAD.EQ.0)GO TO 40                                                 SPLICE    42
      NEXT=NEXTAD                                                             SPLICE    43
   30 CALL OMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                     SPLICE    44
      CALL SUBR(IBD1,NEXT,IVALU)                                              SPLICE    45
   45 IF(IS.EQ.1)CALL OMRLBD(NEXT)                                            SPLICE    46
      IF(NEXTAD.EQ.0)GO TO 40                                                 SPLICE    47
      NEXT=NEXTAD                                                             SPLICE    48
      GO TO 30                                                                SPLICE    49
   40 CONTINUE                                                                SPLICE    50
   50 IF(IS.EQ.1)CALL OMRLBD(IBD1)                                            SPLICE    51
      END                                                                     SPLICE    52
```

```
      SUBROUTINE DISCRVE (ICRVPTR,ISYMBOL,INITSS,NUMDISP,NPLOT)                 ADPTPRE    1
C                                                                               DISCRVE    3
C         THIS ROUTINE DISPLAYS THE CURVE(SET OF POINTS) INDICATED BY           DISCRVE    4
C         "ICRVPTR". THE PRIMARY INPUTS TO THE ROUTINE ARE THE POINTER          DISCRVE    5
C         TO THE CURVE BEAD AND THE PLOT SCALE DATA. PRIMARY OUTPUTS            DISCRVE    6
C         FROM THE ROUTINE ARE:                                                 DISCRVE    7
C              ISYMBOL - THE SYMBOL USED TO DISTINGUISH THIS CURVE FROM         DISCRVE    8
C                        OTHER CURVES ON THE PLOT.                              DISCRVE    9
C               INITSS - THE DISPLAY ADDRESSES FOR THE CURVE ARE STORED         DISCRVE   10
C                        IN ARRAY IDDAD. IDDAD(INITSS) CONTAINS THE             DISCRVE   11
C                        DISPLAY ADDRESS OF THE FIRST DISPLAY REQUIRED          DISCRVE   12
C                        TO EXHIBIT THE CURVE.(INITSS-INITIAL SUBSCRIPT)        DISCRVE   13
C              NUMDISP - NUMDISP CONTAINS THE NUMBER OF DISPLAYS                DISCRVE   14
C                        REQUIRED TO EXHIBIT THE CURVE.                         DISCRVE   15
C              NPLOT   - NPLOT CONTAINS THE NUMBER OF THE                       ADPTPRE    2
C                        PLOT IN THE TABLE                                      ADPTPRE    3
C         INITSS AND NUMDISP ARE NEEDED TO ERASE THE CURVE. THEY WILL BE        DISCRVE   16
C         STORED IN THE CURVE DISPLAY ACCESS TABLE.                             DISCRVE   17
C                                                                               DISCRVE   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                            /AXSCALE/  2
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,          /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                         /CRVBEAD/  3
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),         /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),                  6JULY77    2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM         1SEPT      1
      COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                          CY6777     1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                         6JULY77
      COMMON/PARMSV/XORDGU,YORDGU,DGUPIN                                        /PARMSV/   2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                     /POINTS/   2
      COMMON /SYMTAB/ ISYMTAB(20)                                               /SYMTAB/   2
      DIMENSION MESG(7)                                                         9JULY76    1
C                                                                               DISCRVE   28
C         RECORD THE INITIAL VALUE OF THE IDDAD SUBSCRIPT.                      DISCRVE   29
C                                                                               DISCRVE   30
      INITSS=(NPLOT-1)*4+1                                                      ADPTPRE    4
      NN=INITSS-1                                                               ADPTPRE    5
C                                                                               DISCRVE   32
C         GET THE NUMBER OF POINTS ON THE CURVE, THE CURVE SYMBOL               DISCRVE   33
C         NUMBER, THE RECORD BEAD POINTER, THE X-COORDINATE BEAD                DISCRVE   34
C         POINTER, AND THE Y-COORDINATE BEAD POINTER FROM THE                   DISCRVE   35
C         CURVE BEAD.                                                           DISCRVE   36
C                                                                               DISCRVE   37
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                        DISCRVE   38
      CALL DMGET(ICSYM,ICRVPTR,ISYMBOL)                                         DISCRVE   39
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                         DISCRVE   40
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                          DISCRVE   41
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                          DISCRVE   42
C                                                                               DISCRVE   43
C         IF THE NUMBER OF POINTS EXCEEDS THE PROGRAMMED LIMIT(248),            DISCRVE   44
C         DISPLAY THE FIRST 248 POINTS ONLY.                                    DISCRVE   45
C                                                                               DISCRVE   46
      IF(NPOINTS .LE. 248) GO TO 10                                             DISCRVE   47
      NPOINTS=248                                                               DISCRVE   48
      CALL DMSET(ICNPTS,ICRVPTR,NPOINTS)                                        DISCRVE   49
      CALL ERROR4(0.,7.,62HPOINTS LIMIT EXCEEDED. THE FIRST 248 POINTS W        DISCRVE   50
     XILL BE DISPLAYED.,62,7HDISCRVE,6HINFORM)                                  DISCRVE   51
C                                                                               DISCRVE   52
```

```
      C         CONVERT THE CURVE SYMBOL NUMBER INTO A CURVE SYMBOL.                DISCRVE    53
      C                                                                             DISCRVE    54
   60    10 ISYMBOL = ISYMTAB(ISYMBOL)                                              DISCRVE    55
      C                                                                             DISCRVE    56
      C         MOVE THE CONTENTS OF THE RECORD NUMBER BEAD, THE X-COORDINATE       DISCRVE    57
      C         BEAD AND THE Y-COORDINATE BEAD INTO CORE. THE RECORD NUMBER         DISCRVE    58
      C         BEAD IS A STATUS BEAD FOR THE POINTS IN THE OTHER TWO BEADS.        DISCRVE    59
   65 C              ACTIVE - THE RECORD NUMBER IS POSITIVE OR SHADDED. THE         DISCRVE    60
      C                       RECORD NUMBER IS SHADDED IF AND ONLY IF THE           DISCRVE    61
      C                       POINT HAS BEEN ADDED DURING AN EDIT.                  DISCRVE    62
      C            INACTIVE - THE RECORD NUMBER IS NEGATIVE. A POINT IS           DISCRVE    63
      C                       INACTIVE IF AND ONLY IF IT HAS BEEN DELETED          DISCRVE    64
   70 C                       DURING AN EDIT.                                       DISCRVE    65
      C                                                                             DISCRVE    66
            KOMPCOD = 700000000018 + NPOINTS * 8**6                                 DISCRVE    67
            CALL LMGET(KOMPCOD,IRECPTR,NREC)                                        DISCRVE    68
            CALL DMGET(KOMPCOD,IXPTR,X)                                             DISCRVE    69
   75       CALL DMGET(KOMPCOD,IYPTR,Y)                                            DISCRVE    70
      C                                                                             DISCRVE    71
      C         LOOP THROUGH THE POINTS. IF A POINT IS INACTIVE OR NOT ON THE       DISCRVE    72
      C         PLOT, SKIP IT. OTHERWISE, CONVERT THE COORDINATES TO DGUS AND       DISCRVE    73
      C         PACK THE DESCRIPTION OF THE POINT INTO THE DESCRIPTION BUFFER.      DISCRVE    74
   80 C                                                                             DISCRVE    75
         12 DO 20 I=1,NPOINTS                                                       9JULY76     2
            IF(NREC(I) .LT. 0) GO TO 20                                             DISCRVE    77
            IF ((X(I).LT.XMIN).OR.(X(I).GT.XMAX))GO TO 20                           30JULY76    1
            IF ((Y(I).LT.YMIN).OR.(Y(I).GT.YMAX))GO TO 20                           30JULY76    2
   85       IX=IFIX(((X(I)-XMIN)*DGUPIN)/XINCR+XORDGU-6)                            5OCT        1
            IY=IFIX(((Y(I) - YMIN) * DGUPIN)/YINCR + YORDGU - 11.5)                 DISCRVE    81
            CALL DUPSET(IX,IY,ICODE,IBUF,NBYTE,MBYTE)                               DISCRVE    82
            CALL GUAN(ISYMBOL,1,IBUF,NBYTE,MBYTE)                                   DISCRVE    83
            IF(NBYTE .LT. 512) GO TO 20                                             DISCRVE    84
   90 C                                                                             DISCRVE    85
      C         THE DESCRIPTION BUFFER IS FULL. FLUSH IT.                           DISCRVE    86
      C                                                                             DISCRVE    87
            NN=NN+1                                                                 DISCRVE    88
            CALL GIDISP(NCON,IBUF,NBYTE,IODAD(NN),IGN)                              DISCRVE    89
   95    20 CONTINUE                                                              DISCRVE    90
      C                                                                             DISCRVE    91
      C         ALL POINTS HAVE BEEN CONSIDERED. IS THE DESCRIPTION                 DISCRVE    92
      C         BUFFER EMPTY? IF NOT, FLUSH IT.                                     DISCRVE    93
      C                                                                             DISCRVE    94
  100       IF(NBYTE .EQ. 0) GO TO 30                                             DISCRVE    95
            NN=NN+1                                                                 DISCRVE    96
            CALL GIDISP(NCON,IBUF,NBYTE,IODAD(NN),IGN)                              DISCRVE    97
         30 NUMDISP = NN - INITSS + 1                                               DISCRVE    98
            RETURN                                                                  DISCRVE    99
  105       END                                                                     DISCRVE   100
```

```
      FUNCTION YVA(NP,NDEG,IFIT,ICODE,IV,IER,XX)                        16AUG76   55
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
C     IFIT, 1 LEAST SQUARES, 2 SPLINE, 3 AITKEN                         26AUG      18
C     ICODE, 1 COORD, 2 SLOPES, 3 MIN, 4 DELTA, 5 MAX, 6 TRIM           10MAY77    62
C     IER GT 1 MEANS BAD INTERP                                         16AUG76   59
C     (IER.EQ.6)MEANS ARGUMENT OUT OF RANGE                             29APR      23
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        29APR      24
C                                                                       19APR      1
C     YVA EVALUATES FUNCTION FOR XX, THE CURRENT VALUE OF THE IV        19APR      2
C                                                                       19APR      3
C     IER=0  SUCCESSFUL INTERPOLATION                                   19APR      4
C     IER=1  SUCCESSFUL EXTRAPOLATION                                   19APR      5
C     IER=2  ARGUMENT OUT OF BOUNDS                                     19APR      6
C     IER=3  ATTEMPTING CURVE EVALUATION WITHOUT FIT                    19APR      7
C     IER=4  ATTEMPTING SLOPES CROSSPLOT WITHOUT SPLINE FIT             19APR      8
      IF (IFIT.NE.2) GO TO 120                                          29APR      25
      IF(IV.EQ.1HY)GO TO 105                                            29APR      26
      IF(XX.LT.X(1))GO TO 110                                           29APR      27
      IF(XX.GT.X(NP))GO TO 110                                          29APR      28
      GO TO 120                                                         29APR      29
  105 IF(XX.LT.Y(1))GO TO 110                                           29APR      30
      IF(XX.GT.Y(NP))GO TO 110                                          29APR      31
      GO TO 120                                                         29APR      32
  110 IER=6                                                             29APR      33
      RETURN                                                            29APR      34
  120 CONTINUE                                                          29APR      35
      XV=-4.                                                            19APR      9
      YV=0.                                                             19APR      10
      IF(IFIT.LT.1)GO TO 60                                             16AUG76    62
      IF(IFIT.GT.3)GO TO 60                                             16AUG76    63
      IER=0                                                             16AUG76    64
      GO TO (2,12,30,2,28,2)ICODE                                       10MAY77    63
    2 CONTINUE                                                          16AUG76    66
      GO TO (10,15,35)IFIT                                              16AUG76    67
   10 YY=P(XX,NREC,NP)                                                  16AUG76    68
      GO TO 50                                                          16AUG76    69
   12 NTYP=2                                                            16AUG76    70
      IF(IFIT.NE.1)GO TO 20                                             19APR      11
      YY=PP(XX,NREC,NP)                                                 19APR      12
      GO TO 50                                                          19APR      13
   15 NTYP=1                                                            16AUG76    72
      GO TO 20                                                          16AUG76    73
   20 CONTINUE                                                          26AUG      19
      IF(IFIT.EQ.2)GO TO 21                                             19APR      14
C     SLOPES CROSSPLOT PERMITTED ONLY WITH SPLINE CURVEFIT              19APR      15
      CALL ERROR#(XV,YV-1.,41HSLOPES CROSSPLOT REQUIRES SPLINE CURVEFIT, 19APR      16
     141,3HYVA    ,64HINFORM)                                           19APR      17
      CALL GETSKC(7HQTRBACK)                                            14JUNE     72
   21 CONTINUE                                                          19APR      20
   50 IF(IV.EQ.1HY)GO TO 25                                             26AUG      20
      YY=SPLIN2(XX,NP,X,Y,NREC,NTYP)                                    26AUG      21
      GO TO 59                                                          26AUG      22
   25 YY=SPLIN2(XX,NP,Y,X,NREC,NTYP)                                    26AUG      23
      GO TO 59                                                          16AUG76    75
   28 CONTINUE                                                          10MAY77    64
      IF(IV.NE.1HY)GO TO 29                                             10MAY77    65
      YY=A**X(X,NP)                                                     10MAY77    66
```

```
                      GO TO 50                                                  10MAY77    67
                29    YY=AMAX(Y,NP)                                             10MAY77    68
                      GO TO 50                                                  10MAY77    69
                30    CONTINUE                                                  26AUG      24
                      IF(IV.NE.1HY)GO TO 32                                     20SEPT     152
                      YY=AMIN(X,NP)                                             26AUG      26
                      GO TO 50                                                  26AUG      27
                32    YY=AMIN(Y,NP)                                             26AUG      28
                      GO TO 50                                                  16AUG76    77
                35    CONTINUE                                                  26AUG      29
                      IF(IV.EQ.1HY)GO TO 37                                     26AUG      30
                      CALL INTERP (X,Y,NP,NDEG,XX,YY,NREC,IER)                  LAST8UX    20
                      GO TO 50                                                  26AUG      32
                37    CALL INTERP (Y,X,NP,NDEG,XX,YY,NREC,IER)                  LAST8UX    21
                      GO TO 50                                                  16AUG76    79
                50    YVA=YY                                                    16AUG76    80
                      IF(IER.EQ.0)RETURN                                        19APR      21
                      IF(IER.EQ.1)GO TO 49                                      19APR      22
                      CALL ERROAM(XV,YV-1.,21HARGUMENT OUT OF RANGE  ,21, 3HYVA  ,  19APR      23
                     1 6HINFORM)                                                19APR      24
                      CALL AETSKC(7HQTRBACK)                                    14JUNE     73
                49    CONTINUE                                                  19APR      26
                      RETURN                                                    16AUG76    81
                60    IER=3                                                     19APR      29
                      CALL ERROAM(XV,YV-1.,40HATTEMPTING TO EVALUATE CURVE WITHOUT FIT ,  19APR      30
                     1 40,3HYVA  ,6HINFORM)                                     19APR      31
                      CALL AETSKC(7HQTRBACK)                                    14JUNE     72
                      END                                                       16AUG76
```

```
      FUNCTION PP(Z,C,N)                                                19APR  32
C     COMPUTES DERIVATIVE OF POLYNOMIAL IN Z WITH COEFFICIENTS IN C     19APR  33
C     DEGREE IS N-1                                                     19APR  34
      DIMENSION C(N)                                                    19APR  35
      PP=C(2)                                                           19APR  36
      DO 10 I=3,N                                                       19APR  37
      PP=PP+C(I)*Z**(I-2)                                               19APR  38
   10 CONTINUE                                                          19APR  39
      RETURN                                                            19APR  40
      END                                                               19APR  41
```

```
      SUBROUTINE FITBLK (ICVBD,N,NDEG1,IFIT,IV1)                        FIX2        4
C     IFIT, 1 LSQ, 2 SPLINE, 3 AITKEN                                   16AUG76    17
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG       1
C     J=N*2**16  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/     3
      COMMON/CROSTBL/ICROSS1,ICPSAVE,JP,CTYPE(6),TYPE,JT,YNAME,ICRV1    10MAY77    61
     1,IDELTF,LEELBD(2)                                                 FLCAD2      5
      IFIT=0                                                            16AUG76    20
      CALL DMGET(ICOM1+11,ICVBD,IFITBD)                                 16AUG76    21
      IF(IFITBD.NE.0)GO TO 2                                            5OCT       101
      CALL ERROR1(-3.,3.,34HCURVE FITS REQUIRED FOR CROSSPLOTS ,34,     5OCT       102
     1 6HFITBLK,6HINFORM)                                               5OCT       103
      CALL SETSKC(7HOTRBACK)                                            5OCT       104
    2 CONTINUE                                                          5OCT       105
      CALL DMGET(ICOM1+1,IFITBD,ITYPE)                                  16AUG76    23
      IF(ITYPE.EQ.1HLEASTSQUAR)IFIT=1                                   16AUG76    24
      IF(ITYPE.EQ.6HSPLINE)IFIT=2                                       16AUG76    25
      IF(ITYPE.EQ.6HAITKEN)IFIT=3                                       16AUG76    26
      CALL DMGET(ICOM1+5,IFITBD,N)                                      16AUG76    27
      JJ=N*2**16                                                        16AUG76    28
      CALL DMGET (ICOM1+10,ICVBD,IV1)                                   FIX2        5
      GO TO (10,20,30)IFIT                                              16AUG76    30
   10 CONTINUE                                                          16AUG76    31
      CALL DMGET(ICOM7+JJ+5,IFITBD,NREC)                                16AUG76    32
      IF (JT.NE.3.AND.JT.NE.5) GO TO 11                                 LASTBUX    11
      CALL DMGET (ICOM1+5,ICVBD,N)                                      LASTBUX    12
      JJ=N*2**16                                                        LASTBUX    13
C                                                                       LASTBUX    14
      CALL DMGET (ICOM1+8,ICVBD,IXPTR)                                  LASTBUX    15
      CALL DMGET (ICOM1+9,ICVBD,IYPTR)                                  LASTBUX    16
      CALL DMGET (ICOM7+JJ,IXPTR,X)                                     LASTBUX    17
      CALL DMGET (ICOM7+JJ,IYPTR,Y)                                     LASTBUX    18
   11 CONTINUE                                                          LASTBUX    19
      RETURN                                                            16AUG76    33
   20 CONTINUE                                                          16AUG76    34
      CALL DMGET(ICOM7+JJ+5,IFITBD,NREC)                                16AUG76    35
      KK=JJ+N+5                                                         16AUG76    36
      LL=JJ+2*N+5                                                       16AUG76    37
      CALL DMGET(ICOM7+KK,IFITBD,X)                                     16AUG76    39
      CALL DMGET(ICOM7+LL,IFITBD,Y)                                     16AUG76    40
      RETURN                                                            16AUG76    41
   30 CALL DMGET (ICOM1+6,IFITBD,NDEG1)                                 FIX2        6
      KK=JJ+6                                                           16AUG76    46
      LL=JJ+N+5                                                         16AUG76    47
      CALL DMGET(ICOM7+KK,IFITBD,X)                                     16AUG76    49
      CALL DMGET(ICOM7+LL,IFITBD,Y)                                     16AUG76    50
      RETURN                                                            16AUG76    51
      END                                                               16AUG76    54
```

```
      FUNCTION SPLIN2(X,N,XN,FN,GN,ITYPE)                                SPLIN2     2
C                                                                        SPLIN2     3
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2     4
C                                                                        SPLIN2     5
C                                                                        SPLIN2     6
C - - A FUNCTION UTILITY TO BE USED IN CONJUNCTION WITH THE SPLINE       SPLIN2     7
C     ROUTINES SPLIN4 OR SPLIN1.                                         SPLIN2     8
C                                                                        SPLIN2     9
C         X = THE VALUE OF THE INDEPENDENT VARIABLE AT WHICH THE SPLINE  SPLIN2    10
C             OR ITS DERIVATIVE ARE TO BE EVALUATED. X MUST BE BETWEEN   SPLIN2    11
C             XN(1) AND XN(N) FOR CORRECT RESULTS.                       SPLIN2    12
C         N = THE NUMBER OF KNOTS (RETURNED BY SPLINING ROUTINES)        SPLIN2    13
C        XN = ARRAY OF KNOT POSITIONS (RETURNED BY SPLINE ROUTINES)      SPLIN2    14
C        FN = ARRAY OF SPLINE VALUES (RETURNED BY SPLINE ROUTINES)       SPLIN2    15
C        GN = ARRAY OF DERIVATIVE VALUES (RETURNED BY SPLINE ROUTINES)   SPLIN2    16
C     ITYPE = 1 RETURNS SPLINE VALUE AT X                                SPLIN2    17
C             2 RETURNS THE DERIVATIVE VALUE AT X                        SPLIN2    18
C                                                                        SPLIN2    19
C                                                                        SPLIN2    20
C  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *   SPLIN2    21
C                                                                        SPLIN2    22
      DIMENSION XN(1),FN(1),GN(1)                                        SPLIN2    23
      IF(X .LT. XN(1)) GO TO 100                                         SPLIN2    24
      IF(X .GT. XN(N)) GO TO 110                                         SPLIN2    25
       DO 8 K=1,N                                                        SPLIN2    26
       IF(X .NE. XN(K)) GO TO 8                                          SPLIN2    27
       KK = K                                                            SPLIN2
       GO TO 120                                                         SPLIN2
    8 CONTINUE                                                           SPLIN2    30
      DO 10 K=2,N                                                        SPLIN2    31
      IF(X .GT. XN(K)) GO TO 10                                          SPLIN2    32
      JK=K                                                               SPLIN2    33
      GO TO 20                                                           SPLIN2    34
   10 CONTINUE                                                           SPLIN2    35
   20 HH=XN(JK)-XN(JK-1)                                                 SPLIN2    36
      HH3=HH**3                                                          SPLIN2    37
      T1=HH*GN(JK-1)+FN(JK-1)-FN(JK)                                     SPLIN2    38
      T2=HH*GN(JK)+FN(JK-1)-FN(JK)                                       SPLIN2    39
      T3=XN(JK)-X                                                        SPLIN2    40
      T4=X-XN(JK-1)                                                      SPLIN2    41
      GO TO (200,201),ITYPE                                              SPLIN2    42
  200 SPLIN2=(T3*FN(JK-1)+T4*FN(JK))/HH+T3*T4*(T3*T1-T4*T2)/HH3          SPLIN2    43
      RETURN                                                             SPLIN2    44
  201 SPLIN2=(FN(JK)-FN(JK-1))/HH-T3*T4*(HH*(GN(JK-1)+GN(JK))+2.*(FN(JK- SPLIN2    45
     1        1)-FN(JK)))/HH3+(XN(JK-1)+XN(JK)-2.*X)*(T3*T1-T4*T2)/HH3   SPLIN2    46
      RETURN                                                             SPLIN2    47
  100 JK=2                                                               SPLIN2    48
      GO TO 20                                                           SPLIN2    49
  110 JK=N                                                               SPLIN2    50
      GO TO 20                                                           SPLIN2    51
  120 GO TO (300,301),ITYPE                                              SPLIN2    52
  300 SPLIN2=FN(KK)                                                      SPLIN2    53
      RETURN                                                             SPLIN2    54
  301 SPLIN2=GN(KK)                                                      SPLIN2    55
      RETURN                                                             SPLIN2    56
      END                                                                SPLIN2    57
```

```
      FUNCTION P(Z,C,N)                                                 P          2
C                                                                       P          3
C         THIS ROUTINE CALCULATES THE VALUE AT "Z" OF THE LEAST SQUARES P          4
C         POLYNOMIAL DEFINED BY THE COEFFICIENTS IN ARRAY "C".          P          5
C                                                                       P          6
C         Z - I - THE VALUE OF THE INDEPENDENT VARIABLE.                P          7
C         C - I - THE ARRAY OF LEAST SQUARES POLYNOMIAL COEFFICIENTS.   P          8
C         N - I - THE NUMBER OF COEFFICIENTS.                           P          9
C                                                                       P         10
      DIMENSION C(N)                                                    P         11
      NDEG = N - 1                                                      P         12
      P=C(1)                                                            18JUNE76  21
      DO 10 I=2,N                                                       18JUNE76  22
      P=P+C(I)*Z**(I-1)                                                 18JUNE76  23
   10 CONTINUE                                                          P         16
      RETURN                                                            P         17
      END                                                               P         18
```

```
1                FUNCTION AMIN(A,N)                                    AMIN     2
                 DIMENSION A(N)                                        AMIN     3
                 V=A(1)                                                AMIN     4
                 DO 10 I=2,N                                           AMIN     5
5                IF(A(I).LT.V)V=A(I)                                   AMIN     6
              10 CONTINUE                                              AMIN     7
                 AMIN=V                                                AMIN     8
                 END                                                   AMIN     9
```

```
      SUBROUTINE DELPIC(IPLTPTR,IDELBD)                                  DELPIC      2
      DIMENSION IDELBC(2)                                                DELPIC      3
      COMMON/PARVAL/VALUES(20),IVREAD(20),NVAL,NAME                      DELPIC      4
      COMMON/POINTS/X(248),Y(248),NREC(248)                              /POINTS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG       1
C     J=N*2**18  OMSET(ICOM7+J,IBD4,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/     3
      EXTERNAL PIKK                                                      DELPIC      7
      DIMENSION IPICK(2)                                                 DELPIC      8
      CALL LNGET(ICOM1+11,IPLTPTR,ICRV)                                  DELPIC      9
      NVAL=0                                                             DELPIC     10
      CALL SPLINE(ICRV,PIKK,2,0,1,2,0)                                   DELPIC     11
      IF(NVAL.GE.2)GO TO 5                                               50CT       96
      CALL ERROR*(1.,-3.,58HAT LEAST TWO PARAMETER VALUES REQUIRED FOR D 50CT       97
     1ELTA CROSSPLOT ,58,6HDELPIC,6HINFORM)                              50CT       98
      CALL ABTSKD (7HOTRBACK)                                            LASTBUX     9
    5 CONTINUE                                                           50CT      100
      CALL PICKY(-5.,5.,43HPICK PARAM1,PARAM2,FOR DELTA(PARAM1-PARAM2),  DELPIC     12
     1*,VALUES,NVAL,IPICK,2,NREC,NAME)                                   DELPIC     13
      DO 10 I=1,2                                                        DELPIC     14
   10 IDELBD(I)=IVREAD(IPICK(I))                                         DELPIC     15
      END                                                                DELPIC     16
```

```
1                SUBROUTINE PIKK(IBXX,IBEAD,IVALU)                                  PIKK      2
                 COMMON/PARVAL/VALUES(20),IVBEAD(20),NVAL,NAME                      PIKK      3
                 COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG     1
           C     J=N*2**18  CMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/   3
5                NVAL=NVAL+1                                                        PIKK      5
                 IVBEAD(NVAL)=IBEAD                                                 PIKK      6
                 CALL CMGET(ICOM1+4,IBEAD,VALUES(NVAL))                             PIKK      7
                 CALL CMGET(ICOM1+3,IBEAD,NAME)                                     PIKK      8
                 END                                                                PIKK      9
```

```
      SUBROUTINE PICKY(XOR,YOR,MSSG,NC,VALUES,NVAL,IPICK,NPIC,IDUM,NAME)  PICKY      2
      DIMENSION MSSG(1),VALUES(NVAL),IPICK(NPIC),IDUM(1)                  PICKY      3
      COMMON /GD/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                    CYBER77    1
     X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                   6JULY77    3
C     (XOR,YOR) INCHES. LOCATION OF UPPER LEFT CORNER OF DISPLAY-INPUT     PICKY      5
C     MSSG - ALPHA INSTRUCTION MESSAGE WITH NC CHARACTERS - INPUT          PICKY      6
C     VALUES - AN ARRAY CONTAINING NVAL VALUES - INPUT                     PICKY      7
C     IPICK - AN ARRAY RETURNING THE INDICES OF THE NPIC ITEMS PICKED      PICKY      8
C     IDUM - A SCRATCH ARRAY WITH NVAL+3+2*NPIC LOCATIONS                  PICKY      9
      CALL ORIGIN(XOR,YOR)                                                PICKY     10
      YV=0.                                                               PICKY     11
      YOFF=.5                                                             PICKY     12
      CALL ANNOTA(0.,YV,MSSG,NC,IDUM(NVAL+1),IGN,-0,-0,-0)                PICKY     13
      YV=YV-YOFF                                                          PICKY     14
      CALL ANNOTA(0.,YV,NAME,10,IDUM(NVAL+3),IGN,-0,-0,-0)                PICKY     15
      DO 20 I=1,NVAL                                                      PICKY     16
      ENCODE (10,100,IDUMMY) VALUES(I)                                    20SEPT    146
      YV=YV-YOFF                                                          PICKY     18
   20 CALL ANNOTA(0.,YV,IDUMMY,10,IDUM(I),ISTR,I,-0,-0)                   PICKY     19
      CALL ANNOTA(0.,YV-1.,6HACCEPT,6,IDUM(NVAL+2),IBUT,66,-0,-0)         PICKY     20
    7 CALL GIBUT(0,NCON,IDUTR,IDUCR)                                      PICKY     21
      IF(IDUCR.EQ.66)GO TO 8                                              PICKY     22
      GO TO 7                                                             PICKY     23
    8 NR=NVAL                                                             20SEPT    147
      CALL GIFSID(NCON,NR,IDUM(NVAL+4),IBUF)                              20SEPT    148
      IF(NR.EQ.NPIC)GO TO 9                                               PICKY     25
      CALL ERRORM(0.,YV-2.,13HREPICK VALUES,13,5HPICKY,6HINFORM)          PICKY     26
      GO TO 7                                                             PICKY     27
    9 CONTINUE                                                            PICKY     28
      DO 30 I=1,NR                                                        PICKY     29
   30 IPICK(I)=IBUF(I)                                                    20SEPT    149
      NERAS=NVAL+3                                                        20SEPT    150
      CALL TERASE(IDUM,NERAS)                                             PICKY     32
  100 FORMAT(G10.3)                                                       PICKY     33
      END                                                                 PICKY     34
```

```
      SUBROUTINE SKALBD(IDELB)                                          3FEB    30
      COMMON/TDIS/AA(4),BB(8)                                           3FEB    31
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG    1
C     J=N*2**18  DMSET(ICOM7+J,I8DA,DATA)  STORES DATA(N) IN I8DA(1)    /DISKT/  3
      DATA YL/9./,YL/6./                                                3FEB    33
      DDY=.1*(BB(8)-BB(7))                                              3FEB    34
      BB(3)=YMIN=(BB(7)-DDY)                                            3FEB    35
      YMAX=BB(8)+DDY                                                    3FEB    36
      BB(4)=(YMAX-YMIN)/YL                                              3FEB    37
      CALL DMGET(ICOM1+8,IDELB,ISCLBD)                                  3FEB    38
      JJ=6*2**18                                                        3FEB    39
      CALL DMGTBD(8,ISCXBD)                                             15FEB   10
      CALL DMSET(ICOM7+JJ,ISCXBD,BB)                                    15FEB   11
      CALL DMSET (ICOM1+8,IDELB,ISCXBD)                                 15FEB   12
      END                                                               3FEB    41
```

```
      SUBROUTINE RWPLGR(IBEAD1,INEXT,IPREV,IONE)                        RWPLGR     2
      COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR /BEADS/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      EXTERNAL SUBGET                                                   RWPLGR     5
C     IONE=1 MEANS ONLY ONE PLOT IN GROUP                               RWPLGR     6
      IONE=0                                                            RWPLGR     7
      CALL DMGET(ICOM1+IPREV,IBEAD1,IPR)                                RWPLGR     8
      IF(IPR.NE.0)GO TO 10                                              RWPLGR     9
   10 CALL LMGET(ICOM1+INEXT,IBEAD1,NEXT)                               RWPLGR    10
      IF(NEXT.NE.0)RETURN                                               RWPLGR    11
      IONE=1                                                            RWPLGR    12
      RETURN                                                            RWPLGR    13
C  10 CALL SPLICE(IBEAD1,SUBGET,IPREV,0,IPREV,INEXT,0)                  RWPLGR    14
      END                                                               RWPLGR    15
```

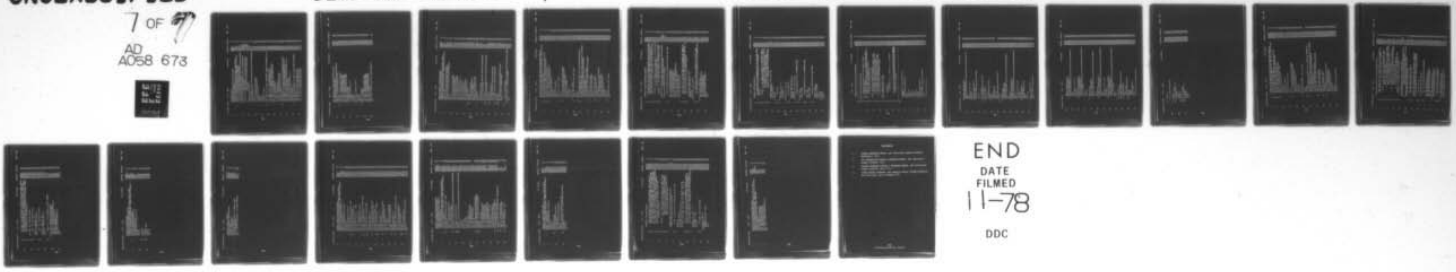
```
1               SUBROUTINE SUBGET(IBEAD1,IBEAD,IVALU)                    RWPLGR     16
                IBEAD1=IBEAD                                             RWPLGR     17
                END                                                      RWPLGR     18
```

```
      SUBROUTINE ITRIMR(IBEAD)                                          ITRIMR     2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/    2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          3CAUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      COMMON/OPTION/IOPTION,ISHINE                                      LASTBUX    1
      CALL DMGET(ICOM1+2,IBEAD,ITRIM)                                   ITRIMR     5
      IF(ITRIM.EQ.0)GO TO 10                                            ITRIMR     6
      IF (IOPTION.NE.4HTRIM) GO TO 10                                   3AUG      16
      INEXT=2                                                           ITRIMR     7
      IPREV=4                                                           ITRIMR     8
C                                                                       ITRIMR     9
C     TRIM MODE                                                         ITRIMR    10
      RETURN                                                            ITRIMR    11
   10 CONTINUE                                                          ITRIMR    12
C     CROSSPLOT MODE                                                    ITRIMR    13
      INEXT=1                                                           ITRIMR    14
      IPREV=3                                                           ITRIMR    15
      END                                                               ITRIMR    16
```

```
      1                 OVERLAY(21,0)                                   DOVER      2
                        PROGRAM DOVER                                   DOVER      3
                        COMMON/SLFIT/ICR12,ICHNG                        /SLFIT/    2
                        CALL PREFIT                                     DOVER      5
      5                 ICHNG=0                                         DOVER      6
                        ISW=ICR12                                       DOVER      7
                        ICR12=3                                         DOVER      8
                        GO TO (10,20)ISW                                DOVER      9
                  10    CALL SETSKC(5HCROSS)                            DOVER     10
     10           20    CALL SETSKC(6HCROSS2)                           DOVER     11
                        END                                             DOVER     12
```

AD-A058 673 DAYTON UNIV OHIO RESEARCH INST F/G 9/2
PLOTTING LANGUAGE UTILIZING GRAPHICAL OPTICS (PLUGO). VOLUME II--ETC(U)
MAY 78 J G JENSEN F33615-76-C-3076
UNCLASSIFIED UDR-TR-78-34 AFFDL-TR-78-55-VOL-2 NL

7 OF 7
AD A058 673



END
DATE FILMED
11-78
DDC

```
      1                SUBROUTINE PREFIT                                                  PREFIT      2
                       COMMON/PRECFIT/IVARS,FTYPE,IDEG,ITYSEL,IVSEL                       PREFIT      3
                       DIMENSION TYPE(3),IVAR(3)                                          PREFIT      4
                       COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG       1
      5          C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/     3
                       COMMON /GF/ IBUF(128),NCON,NBYTE,MBYTE,IGN,ISIN,                   CYBER77     1
                      X     ISTR,IBUT,IMARK,ICODE,IBEAM,ISTYLE,IEOM,CPTO                 6JULY77     3
                       COMMON/BEADS/IPGPTR,IAXPTR,IPMPTR,IPLTPTR,ICRVPTR,IFITPTR,IAXSPTR  /BEADS/     2
                       COMMON/ITRIMT/ITRIM,IAEXT,IPREV,N6                                 /ITRIMT/    2
     10                COMMON/AXSBEAD/IAXSXM,IAXSXI,IAXSY1M,IAXSY1I,IAXSY2M,IAXSY2I,      /AXSBEAD/    2
                      X              IAXSY3M,IAXSY3I                                     /AXSBEAD/    3
                       COMMON/PLTBEAD/IPNEXTC,IPNEXTT,IPPREVC,IPPREVT,IPPG,IPTITLE,       /PLTBEAD/    2
                      X               IPAXIS,IPAXISS,IPPM,IPLABEL,IPCURVE,IPNMULT,       /PLTBEAD/    3
                      X               IPFMN,IPFMV                                        /PLTBEAD/    4
     15                COMMON/SLFIT/ICP12,ICHNG                                           /SLFIT/     2
                       EXTERNAL FPLOT                                                     PREFIT      9
                       DATA TYPE/10HLEASTSQUAR,6HSPLINE,6HAITKEN/,IVAR/1HX,1HY,7HDEFAULT/ 23FEB       6
                       IF(ICHNG.EQ.0)GO TO 10                                             SLOPES     30
                       IVSEL=2                                                            SLOPES     31
     20                CALL DMGET(ICOM1+11,IPLTPTR,ICV3)                                  SLOPES     32
                       CALL DMGET(ICOM1+10,ICV3,IVARS)                                    SLOPES     33
                       IF(IVARS.EQ.1HY)GO TO 5                                            SLOPES     34
                       IVARS=1HX                                                          SLOPES     35
                       IVSEL=1                                                            SLOPES     36
     25            5   CONTINUE                                                           SLOPES     37
                       ITYSEL=2                                                           SLOPES     38
                       FTYPE=TYPE(ITYSEL)                                                 SLOPES     39
                       GO TO 50                                                           SLOPES     40
                  10   CALL ANNOTA(0.,0.,25HPICK INDEPENDENT VARIABLE,25,IOD1,IGN,-0,-0,  PREFIT     13
     30                1-2)                                                               PREFIT     14
                       CALL ANNOTA (0.,-4.,1HX,1,IOD2,ISIN,1,-0,-0)                       3MARCH     40
                       CALL ANNOTA (2.,-4.,1HY,1,IOD3,ISIN,2,-0,-0)                       3MARCH     41
                       CALL ANNOTA (4.,-4.,7HDEFAULT,7,IOD4,ISIN,3,-0,-0)                 3MARCH     42
                       CALL ANNOTA (2.,-6.,6HACCEPT,6,IOD5,IBUT,4,-0,-0)                  3MARCH     43
     35              15 CALL GIBUT (0,NCON,IODT,IVX)                                       3MARCH     44
                        IF (IVX.EQ.254) CALL STOP                                         3MARCH     45
                        IF (IVX.NE.4) GO TO 15                                            3MARCH     46
                        CALL GIFIO (NCON,IODT,IVSEL)                                      3MARCH     47
                        CALL GISRAS (IOD1,IOD2,IOD3,IOD4,IOD5)                            3MARCH     48
     40                 IF(IVSEL.LT.1.OR.IVSEL.GT.3)GO TO 10                              PREFIT     20
                        IVARS=IVAR(IVSEL)                                                 PREFIT     21
                  20    CALL ANNOTA(0.,0.,16HPICK TYPE OF FIT,16,IOD1,IGN,-0,-0,-0)       PREFIT     22
                        CALL ANNOTA(0.,-4.,12HLEAST SQUARE,12,IOD2,IBUT,1,-0,-0)          PREFIT     23
                        CALL ANNOTA (2.,-4.,6HSPLINE,6,IOD3,IBUT,2,-0,-0)                 23FEB       7
     45                 CALL ANNOTA(4.,-4.,13HSTRAIGHT LINE,13,IOD4,IBUT,3,-0,-0)         PREFIT     25
                        CALL GIBUT(0,NCON,IODT,ITYSEL)                                    PREFIT     26
                        CALL GISRAS(IOD1,IOD2,IOD3,IOD4)                                  PREFIT     27
                        IF(ITYSEL.LT.1.OR.ITYSEL.GT.3)GO TO 20                            PREFIT     28
                        FTYPE=TYPE(ITYSEL)                                                PREFIT     29
     50                 IF(ITYSEL.NE.1)GO TO 50                                           PREFIT     30
                  30    CALL ANNOTA(0.,0.,42HSELECT DEGREE OF LEAST SQUARES POLYNOMIAL    PREFIT     31
                       142,IOD1,IGN,-0,-0,-0)                                             PREFIT     32
                        CALL ANNOTA(1.,-4.,1H1,1,IOD2,IBUT,1,-0,-0)                       PREFIT     33
                        CALL ANNOTA(2.,-4.,1H2,1,IOD3,IBUT,2,-0,-0)                       PREFIT     34
     55                 CALL ANNOTA(3.,-4.,1H3,1,IOD4,IBUT,3,-0,-0)                       PREFIT     35
                        CALL ANNOTA(4.,-4.,1H4,1,IOD5,IBUT,4,-0,-0)                       PREFIT     36
                        CALL ANNOTA(5.,-4.,1H5,1,IOD6,IBUT,5,-0,-0)                       PREFIT     37
```

```
      CALL ANNOTA(6.,-4.,1H6,1,IDD7,IBUT,6,-0,-0)                    PREFIT     38
      CALL GIBUT(0,NCON,IDDT,IDEG)                                   PREFIT     39
60    CALL GIERAS(IDD1,IDD2,IDD3,IDD4,IDD5,IDD6,IDD7)                PREFIT     40
      IF(IDEG.LT.1.OR.IDEG.GT.6)GO TO 30                             PREFIT     41
   50 CONTINUE                                                       PREFIT     42
      CALL DMGET (IPAXISS,IPLTPTR,IAXSPTR)                           3MARCH     49
      CALL DMGET (IAXSXM,IAXSPTR,XMIN)                               3MARCH     50
65    CALL DMGET (IAXSXI,IAXSPTR,XINCR)                              3MARCH     51
      CALL DMGET (IAXSY1M,IAXSPTR,YMIN)                              3MARCH     52
      CALL DMGET (IAXSY1I,IAXSPTR,YINCR)                             3MARCH     53
      XMAX=XMIN+9.*XINCR                                             3MARCH     54
      YMAX=YMIN+6.*YINCR                                             3MARCH     55
70    IF (XMIN.LT.XMAX) GO TO 60                                     3MARCH     56
      TEMP=YMIN                                                      3MARCH     57
      XMIN=XMAX                                                      3MARCH     58
      XMAX=TEMP                                                      3MARCH     59
   60 IF (YMIN.LT.YMAX) GO TO 70                                     3MARCH     60
75    TEMP=YMIN                                                      3MARCH     61
      YMIN=YMAX                                                      3MARCH     62
      YMAX=TEMP                                                      3MARCH     63
   70 XINCR=ABS(XINCR)                                               3MARCH     64
      YINCR=ABS(YINCR)                                               3MARCH     65
80    CALL DMSET (IAXSXM,IAXSPTR,XMIN)                               3MARCH     66
      CALL DMSET (IAXSXI,IAXSPTR,XINCR)                              3MARCH     67
      CALL DMSET (IAXSY1M,IAXSPTR,YMIN)                              3MARCH     68
      CALL DMSET (IAXSY1I,IAXSPTR,YINCR)                             3MARCH     69
      CALL ITRIMR(IPLTPTR)                                           PREFIT
85    CALL SPLICE (IPLTPTR,PPLOT,12,1,INEXT,IPREV,0)                 14 JUNE
      END                                                            PREFIT     45
```

```
      1                 SUBROUTINE PCURVE2 (IBEAD1,IBEAD,IVALU)                          PCURVE2     2
                        COMMON/POINTS/X(249),Y(248),NREC(248)                            /POINTS/    2
                        COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG         30AUG       1
                  C     J=2**18  CMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/     3
      5                 INTEGER FTYPE                                                    23FEB       8
                        COMMON/PRECFIT/IVARS,FTYPE,IDEG,ITYSEL,IVSEL                     PCURVE2     5
                        DIMENSION W(742),DERIV(248),WORK(42),POLYCF(7)                   PCURVE2     6
                        EQUIVALENCE (DERIV(1),POLYCF(1)),(W(249),WORK(1))                PCURVE2     7
                        IEN=0                                                            NEWFACE    31
     10                 IV=IVARS                                                         PCURVE2     8
                        CALL CMGET(ICOM1+5,IBEAD,NPOINTS)                                23FEB       9
                        CALL CMGET(ICOM1+12,IBEAD,ISTAT)                                 PCURVE2     9
                        IF(ISTAT.EQ.8HINACTIVE)RETURN                                    PCURVE2    10
                        CALL CMGET(ICOM1+10,IBEAD,IVOLD)                                 PCURVE2    11
     15                 CALL CMGET(ICOM1+11,IBEAD,IFIT)                                  PCURVE2    12
                        IF(IFIT.EQ.0)GO TO 20                                            PCURVE2    13
                        CALL CMGET(ICOM1+1,IFIT,OTYPE)                                   PCURVE2    14
                        IF(FTYPE.NE.OTYPE)GO TO 9                                        PCURVE2    15
                        IF(IVOLD.NE.IVARS)GO TO 9                                        PCURVE2    16
     20                 IF(ITYSEL.NE.1)RETURN                                            PCURVE2    17
                        CALL CMGET(ICOM1+5,IFIT,NCOF)                                    PCURVE2    18
                        IF(IDEG+1.EQ.NCOF)RETURN                                         PCURVE2    19
                    9   CALL CMRLSE(IFIT)                                               PCURVE2    20
                   20   CONTINUE                                                         PCURVE2    21
     25                 IF(IVSEL.EQ.3)IV=IVOLD                                           PCURVE2    22
                        CALL SORTCK(IBEAD,IV,NPOINTS)                                    PCURVE2    23
                        CALL MRCLEAN (NPOINTS)                                           25JULY77   14
                        IF(FTYPE.EQ.6HWAITKEN)GO TO 295                                  23FEB      10
                        IF(FTYPE.EQ.6HSPLINE)GO TO 200                                   PCURVE2    25
     30             C   LEAST SQUARE                                                    PCURVE2    26
                        NEP=IDEG+1                                                       PCURVE2    27
                        IN=0                                                             23FEB      11
                        IF(IV.EQ.1HY)GO TO 100                                           PCURVE2    28
                        CALL PLSCF(X,Y,W,NPOINTS,IDEG,NMX,POLYCF,IN,XD,X0,WORK,IER)      23FEB      12
     35                 IF (IER.NE.0) GO TO 350                                          3MARCH     70
                        GO TO 300                                                        23FEB      13
                   100  CALL PLSCF(Y,X,W,NPOINTS,IDEG,NMX,POLYCF,IN,XD,X0,WORK,IER)      23FEB      14
                        IF (IER.NE.0) GO TO 350                                          3MARCH     71
                        GO TO 300                                                        PCURVE2    32
     40                 200  NEP=NPOINTS                                                PCURVE2    33
                        IF(IV.EQ.1HY)GO TO 205                                           PCURVE2    34
                        CALL SPLINC(NPOINTS,X,Y,DERIV,W,0,DUM1,DUM2)                     PCURVE2    35
                        GO TO 300                                                        PCURVE2    36
                   205  CALL SPLINC(NPOINTS,Y,X,DERIV,W,0,DUM1,DUM2)                     PCURVE2    37
     45                 GO TO 300                                                        23FEB      15
                   295  NEP=IDEG+1                                                       PCURVE2    38
                   300  CONTINUE                                                         PCURVE2    39
                        CALL BUILDFB(FTYPE,DERIV,NEP,IV,NPOINTS,ICRVFIT)                 PCURVE2    40
                        CALL CMSET(ICOM1+11,IBEAD,ICRVFIT)                               PCURVE2    41
     50                 RETURN                                                           3MARCH     72
                  C                                                                      3MARCH     73
                  C       ERROR - LEAST SQUARES                                          3MARCH     74
                    350 CALL ERRORM (0.,-5.,6HERRORS IN LEAST SQUARES--POSSIBLE BAD INDEP 3MARCH     75
                       1ENDENT VARIABLE, 66,7HPCURVE2,6HINFORM)                          3MARCH     76
     55                 RETURN                                                           3MARCH     77
                        END                                                              PCURVE2    42
```

```
      SUBROUTINE SPLICE(IBD1,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)                  SPLICE     2
C     SPLICE SEARCHES STRING BEGINNING WITH IBD1 FOR A MATCH TO VALU            SPLICE     3
C     IN WORD JMATCH OF EACH BEAD.                                              SPLICE     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND                     SPLICE     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                            SPLICE     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                        SPLICE     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                                    SPLICE     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                        SPLICE     9
      DATA ICOMP/01000000000000/                                                SPLICE    10
   10 NEXT=IBD1                                                                 SPLICE    11
      IPREV=IBD1                                                                SPLICE    12
      IF(IBD1.NE.0)GOTO10                                                       SPLICE    13
      PRINT5                                                                    SPLICE    14
    5 FORMAT(17H IBD1 IS NULL )                                                 SPLICE    15
      IS=2                                                                      SPLICE    16
      RETURN                                                                    SPLICE    17
   15 CALL DMSET(ICOMP+JMATCH,NEXT,ICKVALU)                                     SPLICE    18
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                       SPLICE    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                               SPLICE    20
      IF(NEXTAD.EQ.0)GOTO12                                                     SPLICE    21
      IPREV=NEXT                                                                SPLICE    22
      NEXT=NEXTAD                                                               SPLICE    23
      GO TO 10                                                                  SPLICE    24
   12 WRITE (6,6) IVALU                                                         CY8777    21
    6 FORMAT(19H NO MATCH FOR IVALU ,O20)                                       SPLICE    26
      IS=2                                                                      SPLICE    27
      RETURN                                                                    SPLICE    28
   15 CONTINUE                                                                  SPLICE    29
C     IF NEXT EQUALS IBD1 DO SUBR FOR ALL BEADS AND RELEASE ALL                 SPLICE    30
      IF(NEXT.EQ.IBD1)GO TO 25                                                  SPLICE    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE             SPLICE    32
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                           SPLICE    33
      IF(NEXTAD.EQ.0)GOTO22                                                     SPLICE    34
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                       SPLICE    35
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                          SPLICE    36
   22 CONTINUE                                                                  SPLICE    37
      CALL SUBR(IBD1,NEXT,IVALU)                                                SPLICE    38
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                              SPLICE    39
      RETURN                                                                    SPLICE    40
   25 CALL SUBR(IBD1,IBD1,IVALU)                                                SPLICE    41
      IF(NEXTAD.EQ.0)GO TO 40                                                   SPLICE    42
      NEXT=NEXTAD                                                               SPLICE    43
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                       SPLICE    44
      CALL SUBR(IBD1,NEXT,IVALU)                                                SPLICE    45
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                              SPLICE    46
      IF(NEXTAD.EQ.0)GO TO 40                                                   SPLICE    47
      NEXT=NEXTAD                                                               SPLICE    48
      GO TO 30                                                                  SPLICE    49
   40 CONTINUE                                                                  SPLICE    50
      IF(IS.EQ.1)CALL DMRLBD(IBD1)                                              SPLICE    51
      END                                                                       SPLICE    52
```

```
      SUBROUTINE SORTCK(ICRVPTR,IV,NPOINTS)                                  SORTCK     2
C                                                                            SORTCK     3
C         THIS ROUTINE MOVES THE COORDINATE ARRAYS AND THE STATUS ARRAY      SORTCK     4
C         FROM THE DATA MANAGER FILE INTO CORE AND CHECKS TO SEE IF THE      SORTCK     5
C         POINTS HAVE TO BE SORTED. IF SO, THEN THEY ARE SORTED AND          SORTCK     6
C         MOVED BACK INTO THE DATA MANAGER FILE IN THEIR NEW SORT ORDER.     SORTCK     7
C                                                                            SORTCK     8
C         ICRVPTR - I - THE POINTER TO THE CURVE BEAD OF THE CURVE WHICH     SORTCK     9
C                       IS TO BE EXAMINED FOR SORT ORDER.                    SORTCK    10
C              IV - I - THE INDEPENDENT VARIABLE SUBMITTED BY THE USER.      SORTCK    11
C         NPOINTS - O - THE NUMBER OF POINTS IN THE COORDINATE BEADS.        SORTCK    12
C                                                                            SORTCK    13
      COMMON/CRVBEAD/ICNEXT,ICPREV,ICPARAN,ICPARAV,ICNPTS,ICSYM,ICREC,       /CRVBEAD/  2
     X               ICXPTR,ICYPTR,ICINDEP,ICFIT,ICSTAT                      /CRVBEAD/  3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/   2
C                                                                            SORTCK    16
C         MOVE THE COORDINATE ARRAYS AND THE STATUS ARRAY INTO CORE.         SORTCK    17
C                                                                            SORTCK    18
      CALL DMGET(ICNPTS,ICRVPTR,NPOINTS)                                     SORTCK    19
      CALL DMGET(ICREC,ICRVPTR,IRECPTR)                                      SORTCK    20
      CALL DMGET(ICXPTR,ICRVPTR,IXPTR)                                       SORTCK    21
      CALL DMGET(ICYPTR,ICRVPTR,IYPTR)                                       SORTCK    22
      KOMPCOD = 700000000018 + SHIFT(NPOINTS,18)                             SORTCK    23
      CALL DMGET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    24
      CALL DMGET(KOMPCOD,IXPTR,X)                                            SORTCK    25
      CALL DMGET(KOMPCOD,IYPTR,Y)                                            SORTCK    26
C                                                                            SORTCK    27
C         CHECK TO SEE IF THE POINTS NEED TO BE SORTED. THEY SHOULD BE       SORTCK
C         SORTED WITH THE SORT KEY EQUAL TO THE INDEPENDENT VARIABLE         SORTCK    29
C         SUBMITTED BY THE USER. IF THEY ARE ALREADY SORTED THAT WAY, NO     SORTCK    30
C         SORT IS NECESSARY. IF NOT, THEY MUST BE SORTED.                    SORTCK    31
C                                                                            SORTCK    32
      CALL DMGET(ICINDEP,ICRVPTR,IVAR)                                       SORTCK    33
      IF(IVAR .NE. 0) GO TO 10                                               SORTCK    34
      GO TO 20                                                               29APR     18
   10 IF(IV .EQ. IVAR) RETURN                                                SORTCK    37
C                                                                            SORTCK    38
C         THE POINTS ARE NOT SORTED AS SPECIFIED BY THE USER. SORT           SORTCK    39
C         THEM AND DUPLICATE THE SORTED ARRAYS IN THE DATA MANAGER FILE.     SORTCK    40
C                                                                            SORTCK    41
   20 CALL SORTER (NPOINTS,IV)                                               29APR     19
      CALL DMSET(KOMPCOD,IRECPTR,NREC)                                       SORTCK    43
      CALL DMSET(KOMPCOD,IXPTR,X)                                            SORTCK    44
      CALL DMSET(KOMPCOD,IYPTR,Y)                                            SORTCK    45
      CALL DMSET(ICINDEP,ICRVPTR,IV)                                         SORTCK    46
      RETURN                                                                 SORTCK    47
      END                                                                    SORTCK    48
```

```
      SUBROUTINE SORTER(NPOINTS,IV)                                     SORTER     2
C                                                                       SORTER     3
C         THIS ROUTINE SORTS THE COORDINATE ARRAYS AND THE STATUS ARRAY. SORTER     4
C         THE SORT KEY IS THE ARRAY SPECIFIED BY THE INDEPENDENT        SORTER     5
C         VARIABLE.                                                     SORTER     6
C                                                                       SORTER     7
C         NPOINTS - I - THE NUMBER OF POINTS IN THE COORDINATE ARRAYS.  SORTER     8
C              IV - I - THE INDEPENDENT VARIABLE. IF IV = 1HX, THEN THE SORTER     9
C                   THREE ARRAYS (X,Y, AND NREC) ARE SORTED SO THAT THE SORTER    10
C                   X - COORDINATE ARRAY IS IN ASCENDING ORDER.         SORTER    11
C                   IF IV = 1HY, THEN THE THREE ARRAYS ARE SORTED SO    SORTER    12
C                   THAT THE Y - COORDINATE ARRAY IS IN ASCENDING ORDER. SORTER    13
C                                                                       SORTER    14
      COMMON/POINTS/X(248),Y(248),NREC(248)                             /POINTS/   2
      DO 10 L=1,NPOINTS,L                                               SORTER    16
      M = 2 * L - 1                                                     SORTER    17
   10 CONTINUE                                                          SORTER    18
C                                                                       SORTER    19
C         START LOOP                                                    SORTER    20
C                                                                       SORTER    21
   20 M = M / 2                                                         SORTER    22
      IF(M .EQ. 0) GO TO 70                                             SORTER    23
      K = NPOINTS - M                                                   SORTER    24
      DO 60 J=1,K                                                       SORTER    25
      L = J                                                             SORTER    26
   30 IF(L .LT. 1) GO TO 60                                             SORTER    27
C                                                                       SORTER    28
C         USE THE SORT KEY.                                             SORTER    29
C                                                                       SORTER    30
      IF(IV .EQ. 1HY) GO TO 40                                          SORTER    31
      IF(X(L+M) .GE. X(L)) GO TO 60                                     SORTER    32
      GO TO 50                                                          SORTER    33
   40 IF(Y(L+M) .GE. Y(L)) GO TO 60                                     SORTER    34
C                                                                       SORTER    35
C         EXCHANGE POSITIONS IN THE X - COORDINATE ARRAY.               SORTER    36
C                                                                       SORTER    37
   50 TEMP = X(L)                                                       SORTER    38
      X(L) = X(L+M)                                                     SORTER    39
      X(L+M) = TEMP                                                     SORTER    40
C                                                                       SORTER    41
C         EXCHANGE POSITIONS IN THE Y - COORDINATE ARRAY.               SORTER    42
C                                                                       SORTER    43
      TEMP = Y(L)                                                       SORTER    44
      Y(L) = Y(L+M)                                                     SORTER    45
      Y(L+M) = TEMP                                                     SORTER    46
C                                                                       SORTER    47
C         EXCHANGE POSITIONS IN THE STATUS ARRAY.                       SORTER    48
C                                                                       SORTER    49
      ITEMP = NREC(L)                                                   SORTER    50
      NREC(L) = NREC(L+M)                                               SORTER    51
      NREC(L+M) = ITEMP                                                 SORTER    52
      L=L-M                                                             8JUNE76   15
      GO TO 30                                                          SORTER    54
   60 CONTINUE                                                          SORTER    55
      GO TO 20                                                          SORTER    56
   70 RETURN                                                            SORTER    57
      END                                                               SORTER    58
```

```
      SUBROUTINE SPLINC(N,XN,FN,GN,W,ISW,RR1,RRN)                        SPLINC     2
C                                                                        SPLINC     3
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *     SPLINC     4
C                                                                        SPLINC     5
C  CUBIC SPLINE INTERPOLATION ROUTINE.                                   SPLINC     6
C                                                                        SPLINC     7
C    N   = NUMBER OF DATA POINTS (I.E. THE NUMBER OF KNOTS)              SPLINC     8
C    XN  = ARRAY OF X-VALUES                                             SPLINC     9
C    FN  = ARRAY OF FUNCTION VALUES. (I.E. REQUIRED SPLINE VALUES)       SPLINC    10
C    GN  = ARRAY CONTAINING THE VALUES OF THE FIRST DERIVATIVE OF THE    SPLINC    11
C          SPLINE AT EACH KNOT. (RETURNED BY SPLINC)                     SPLINC    12
C    W   = A WORKING ARRAY DIMENSIONED 3*N-2                             SPLINC    13
C                                                                        SPLINC    14
C    ISW = 0 NO BOUNDARY CONDITIONS SPECIFIED                            SPLINC    15
C        = 1 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(1)          SPLINC    16
C        = 2 USER SUPPLIES SECOND DERIVATIVE OF SPLINE AT XN(N)          SPLINC    17
C        = 3 USER SUPPLIES BOTH SECOND DERIVATIVE VALUES                 SPLINC    18
C        = 4 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(1)           SPLINC    19
C        = 5 USER SUPPLIES FIRST DERIVATIVE OF SPLINE AT XN(N)           SPLINC    20
C        = 6 USER SUPPLIES BOTH FIRST DERIVATIVE VALUES                  SPLINC    21
C                                                                        SPLINC    22
C    RR1 = BOUNDARY VALUE AT XN(1)                                       SPLINC    23
C    RRN = BOUNDARY VALUE AT XN(N)                                       SPLINC    24
C                                                                        SPLINC    25
C                                                                        SPLINC    26
C        CONDITIONS:                                                     SPLINC    27
C          1  N .GE. 3                                                   SPLINC    28
C          2  MUST HAVE XN(1) .LT. XN(2) .LT. ... .LT. XN(N)             SPLINC    29
C                                                                        SPLINC    30
C                                                                        SPLINC    31
C        THE RESULTS OF SPLINC CAN BE COUPLED WITH SPLIN2 TO EVALUATE    SPLINC    32
C        THE SPLINE OR ITS FIRST DERIVATIVE FOR ANY X WHICH SATISFIES    SPLINC    33
C          XN(1) .GE. X .LE. XN(N)                                       SPLINC    34
C                                                                        SPLINC    35
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *     SPLINC    36
C                                                                        SPLINC    37
C                                                                        SPLINC    38
      DIMENSION W(1),XN(1) ,FN(1),GN(1)                                  SPLINC    39
      IK=ISW+1                                                           SPLINC    40
      GO TO(1,2,3,4,60,70,80),IK                                         SPLINC    41
    1 R1=0.                                                              SPLINC    42
      RN=0.                                                              SPLINC    43
      GO TO 11                                                           SPLINC    44
    2 R1=RR1                                                             SPLINC    45
      RN=0.                                                              SPLINC    46
      GO TO 11                                                           SPLINC    47
    3 R1=0.                                                              SPLINC    48
      RN=RRN                                                             SPLINC    49
      GO TO 11                                                           SPLINC    50
    4 R1=RR1                                                             SPLINC    51
      RN=RRN                                                             SPLINC    52
   11 W(2)=1.0/(XN(2)-XN(1))                                             SPLINC    53
      W(1)=2.*W(2)                                                       SPLINC    54
      GN(1)=3.*(FN(2)-FN(1))*W(2)**2-R1/2.                               SPLINC    55
      K=2                                                                SPLINC    56
      J=2                                                                SPLINC    57
      W(K)=1.0/(XN(J)-XN(J-1))                                           SPLINC    58
```

```
      5 W(K+2)=1.0/(XN(J+1)-XN(J))                                          SPLINC     59
        W(K+1)=2.*(W(K)+W(K+2))                                             SPLINC     60
60      GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2       SPLINC     61
        K=K+3                                                               SPLINC     62
        J=J+1                                                               SPLINC     63
        IF(J .EQ. N  ) GO TO 8                                              SPLINC     64
        W(K)=W(K-1)                                                         SPLINC     65
65      GO TO 5                                                             SPLINC     66
      8 W(K)=1.0/(XN(N)-XN(N-1))                                            SPLINC     67
        W(K+1)=2.*W(K)                                                      SPLINC     68
        GN(J)=3.*(FN(N)-FN(N-1))*W(K)**2+RN/2.                              SPLINC     69
C                                                                           SPLINC     70
70      MA=3*N-2                                                            SPLINC     71
C                                                                           SPLINC     72
        DO 10 J=4,MA,3                                                      SPLINC     73
     10 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                                    SPLINC     74
        DO 20 J=3,MA,3                                                      SPLINC     75
75   20 W(J)=W(J)/W(J-2)                                                    SPLINC     76
        IC=3                                                                SPLINC     77
        DO 30 J=2,N                                                         SPLINC     78
        GN(J)=GN(J)-W(IC)*GN(J-1)                                           SPLINC     79
     30 IC=IC+3                                                             SPLINC     80
80      N1=N+1                                                              SPLINC     81
        IC=MA-2                                                             SPLINC     82
        GN(N)=GN(N)/W(MA)                                                   SPLINC     83
        DO 40 J=2,N                                                         SPLINC     84
        K=N1-J                                                              SPLINC
85      GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                                 SPLINC
     40 IC=IC-3                                                             SPLINC     87
C                                                                           SPLINC     88
        RETURN                                                              SPLINC     89
     60 H=XN(2)-XN(1)                                                       SPLINC     90
90      W(2)=1./(XN(3)-XN(2))                                               SPLINC     91
        W(1)=2.*(W(2)+1.0/H)                                                SPLINC     92
        GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H          SPLINC     93
        K=3                                                                 SPLINC     94
        J=2                                                                 SPLINC     95
95      W(K)=W(K-1)                                                         SPLINC     96
     61 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                        SPLINC     97
        W(K+1)=2.0*(W(K)+W(K+2))                                            SPLINC     98
        GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2     SPLINC     99
        J=J+1                                                               SPLINC    100
100     K=K+3                                                               SPLINC    101
        IF(J .EQ. N-1) GO TO 62                                             SPLINC    102
        W(K)=W(K-1)                                                         SPLINC    103
        GO TO 61                                                            SPLINC    104
     62 W(K)=XN(N)-XN(N-1)                                                  SPLINC    105
105     W(K+1)=2.*W(K)                                                      SPLINC    106
        GN(J)=3.*(FN(N)-FN(N-1))                                            SPLINC    107
        AN=N-1                                                              SPLINC    108
        IV=1                                                                SPLINC    109
        GO TO 200                                                           SPLINC    110
110   C                                                                     SPLINC    111
     70 W(2)=1.0/(XN(2)-XN(1))                                              SPLINC    112
        W(1)=2.*W(2)                                                        SPLINC    113
        GN(1)=3.*(FN(2)-FN(1))*W(2)**2                                      SPLINC    114
        K=3                                                                 SPLINC    115
```

```
      J=2                                                                SPLINC   116
      W(K)=1.0/(XN(J)-XN(J-1))                                           SPLINC   117
   71 W(K+2)=1.0/(XN(J+1)-XN(J))                                         SPLINC   118
      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   119
      GN(J)=3.*(FN(J)-FN(J-1))*W(K)**2+3.*(FN(J+1)-FN(J))*W(K+2)**2      SPLINC   120
      K=K+3                                                              SPLINC   121
      J=J+1                                                              SPLINC   122
      IF(J .EQ. N-1) GO TO 72                                            SPLINC   123
      W(K)=W(K-1)                                                        SPLINC   124
      GO TO 71                                                           SPLINC   125
   72 W(K)=1.0/(XN(N-1)-XN(N-2))                                         SPLINC   126
      W(K+2)=1.0/(XN(N)-XN(N-1))                                         SPLINC   127
      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   128
      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2    SPLINC   129
     1     -RR*W(K+2)                                                    SPLINC   130
      NN=N-1                                                             SPLINC   131
      IK=2                                                               SPLINC   132
      GO TO 200                                                          SPLINC   133
C                                                                        SPLINC   134
   80 H=XN(2)-XN(1)                                                      SPLINC   135
      W(2)=1.0/(XN(3)-XN(2))                                             SPLINC   136
      W(1)=2.0*(W(2)+1.0/H)                                              SPLINC   137
      GN(1)=3.*(FN(2)-FN(1))/H**2+3.*W(2)**2*(FN(3)-FN(2))-RR1/H         SPLINC   138
      K=3                                                                SPLINC   139
      J=2                                                                SPLINC   140
      W(K)=W(K-1)                                                        SPLINC   141
   81 W(K+2)=1.0/(XN(J+2)-XN(J+1))                                       SPLINC   142
      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   143
      GN(J)=3.*(FN(J+1)-FN(J))*W(K)**2+3.*(FN(J+2)-FN(J+1))*W(K+2)**2    SPLINC   144
      J=J+1                                                              SPLINC   145
      K=K+3                                                              SPLINC   146
      IF(J .EQ. N-2) GO TO 82                                            SPLINC   147
      W(K)=W(K-1)                                                        SPLINC   148
      GO TO 81                                                           SPLINC   149
   82 W(K)=1.0/(XN(N-1)-XN(N-2))                                         SPLINC   150
      W(K+2)=1.0/(XN(N)-XN(N-1))                                         SPLINC   151
      W(K+1)=2.*(W(K)+W(K+2))                                            SPLINC   152
      GN(J)=3.*(FN(N-1)-FN(N-2))*W(K)**2+3.*(FN(N)-FN(N-1))*W(K+2)**2    SPLINC   153
     1     -RR*W(K+2)                                                    SPLINC   154
      NN=N-2                                                             SPLINC   155
      IK=3                                                               SPLINC   156
C                                                                        SPLINC   157
  200 MA=3*NN-2                                                          SPLINC   158
      DO 210 J=4,MA,3                                                    SPLINC   159
  210 W(J)=W(J)-(W(J-1)*W(J-2)/W(J-3))                                   SPLINC   160
      DO 220 J=3,MA,3                                                    SPLINC   161
  220 W(J)=W(J)/W(J-2)                                                   SPLINC   162
      IC=3                                                               SPLINC   163
      DO 230 J=2,NN                                                      SPLINC   164
      GN(J)=GN(J)-W(IC)*GN(J-1)                                          SPLINC   165
  230 IC=IC+3                                                            SPLINC   166
      N1=NN+1                                                            SPLINC   167
      IC=MA-2                                                            SPLINC   168
      GN(NN)=GN(NN)/W(MA)                                                SPLINC   169
      DO 240 J=2,NN                                                      SPLINC   170
      K=N1-J                                                             SPLINC   171
      GN(K)=(GN(K)-GN(K+1)*W(IC))/W(IC-1)                                SPLINC   172
```

```
      240 IC=IC-3                                                       SPLINC  173
C                                                                       SPLINC  174
      GO TO (300,320,340),IK                                            SPLINC  175
C                                                                       SPLINC  176
      300 N1=N+1                                                        SPLINC  177
      DO 305 J=1,NN                                                     SPLINC  178
      305 GN(N1-J)=GN(N1-J-1)                                           SPLINC  179
      GN(1)=RR1                                                         SPLINC  180
      RETURN                                                            SPLINC  181
      320 GN(N)=RRN                                                     SPLINC  182
      RETURN                                                            SPLINC  183
      340 DO 345 J=1,NN                                                 SPLINC  184
      345 GN(N-J)=GN(N-J-1)                                             SPLINC  185
      GN(1)=RR1                                                         SPLINC  186
      GN(N)=RRN                                                         SPLINC  187
      RETURN                                                            SPLINC  188
      END                                                               SPLINC  189
```

575

```
      SUBROUTINE SPLICE2(IB01,SUBR,JMATCH,IVALU,JNEXT,JPREV,IS)          SPLICE2     2
C     SPLICE SEARCHES STRING BEGINNING WITH IB01 FOR A MATCH TO VALU     SPLICE2     3
C     IN WORD JMATCH OF EACH BEAD.                                       SPLICE2     4
C     WHEN THIS BEAD IS FOUND PROCEDURE SUB IS EXECUTED AND              SPLICE2     5
C     THE BEAD WITH THE MATCH TO VALU IS SPLICED OUT                     SPLICE2     6
C     IF(IS.NE.1)BEAD IS NOT SPLICED OUT                                 SPLICE2     7
C     POINTER TO PREVIOUS BEAD IN WORD JPREV                             SPLICE2     8
C     POINTER TO NEXT BEAD IN WORD JNEXT                                 SPLICE2     9
      DATA ICOMP/O100000000000/                                          SPLICE2    10
      NEXT=IB01                                                          SPLICE2    11
      IPREV=IB01                                                         SPLICE2    12
      IF(IB01.NE.0)GOTO10                                                SPLICE2    13
      WRITE (6,5)                                                        CYB777     22
    5 FORMAT(13H IB01 IS NULL )                                          SPLICE2    15
      IS=2                                                               SPLICE2    16
      RETURN                                                             SPLICE2    17
   10 CALL DMGET(ICOMP+JMATCH,NEXT,ICKVALU)                              SPLICE2    18
      CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                SPLICE2    19
      IF(ICKVALU.EQ.IVALU) GOTO15                                        SPLICE2    20
      IF(NEXTAD.EQ.0)GOTO12                                              SPLICE2    21
      IPREV=NEXT                                                         SPLICE2    22
      NEXT=NEXTAD                                                        SPLICE2    23
      GO TO 10                                                           SPLICE2    24
   12 WRITE (6,6) IVALU                                                  CYB777     23
    6 FORMAT(19H NO MATCH FOR IVALU ,O20)                                SPLICE2    26
      IS=2                                                               SPLICE2    27
      RETURN                                                             SPLICE2
   15 CONTINUE                                                           SPLICE2
C     IF NEXT EQUALS IB01 DO SUBR FOR ALL BEADS AND RELEASE ALL          SPLICE2    30
      IF(NEXT.EQ.IB01)GO TO 25                                           SPLICE2    31
C     OTHERWISE DO SUBR FOR BEAD WITH MATCH, SPLICE OUT AND RELEASE      SPLICE2    32
      IF(IS.EQ.1)CALL DMSET(ICOMP+JNEXT,IPREV,NEXTAD)                    SPLICE2    33
      IF(NEXTAD.EQ.0)GOTO22                                              SPLICE2    34
      CALL DMGET(ICOMP+JPREV,NEXT,IPREAD)                                SPLICE2    35
      IF(IS.EQ.1)CALL DMSET(ICOMP+JPREV,NEXTAD,IPREAD)                   SPLICE2    36
   22 CONTINUE                                                           SPLICE2    37
      CALL SUBR(IB01,NEXT,IVALU)                                         SPLICE2    38
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    39
      RETURN                                                             SPLICE2    40
   25 CALL SUBR(IB01,IB01,IVALU)                                         SPLICE2    41
      IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    42
      NEXT=NEXTAD                                                        SPLICE2    43
   30 CALL DMGET(ICOMP+JNEXT,NEXT,NEXTAD)                                SPLICE2    44
      CALL SUBR(IB01,NEXT,IVALU)                                         SPLICE2    45
      IF(IS.EQ.1)CALL DMRLBD(NEXT)                                       SPLICE2    46
      IF(NEXTAD.EQ.0)GO TO 40                                            SPLICE2    47
      NEXT=NEXTAD                                                        SPLICE2    48
      GO TO 30                                                           SPLICE2    49
   40 CONTINUE                                                           SPLICE2    50
      IF(IS.EQ.1)CALL DMRLBD(IB01)                                       SPLICE2    51
      END                                                                SPLICE2    52
```

```
      SUBROUTINE BUILDF8(ITYPE,A,NFP,IV,NPOINTS,IFITPTR)                BUILDF8    2
C                                                                       BUILDF8    3
C         THIS ROUTINE GETS A FIT BEAD AND FILLS IT WITH INFORMATION    BUILDF8    4
C         ABOUT THE FIT AND THE INTERVAL OVER WHICH IT APPLIES.         BUILDF8    5
C                                                                       BUILDF8    6
C           ITYPE - I - THE TYPE OF FIT; 10HLEASTSQUAR FOR A LEAST      BUILDF8    7
C                   SQUARES FIT, 6HSPLINE FOR A SPLINE FIT, AND         13SEPT    12
C                   6HAITKEN FOR A STRAIGHT FIT.                        13SEPT    13
C               A - I - THE ARRAY OF FIT PARAMETERS; THE LEAST SQUARES  BUILDF8    9
C                   POLYNOMIAL COEFFICIENTS FOR A LEAST SQUARES FIT AND BUILDF8   10
C                   THE ARRAY OF DERIVATIVES FOR A SPLINE FIT.          BUILDF8   11
C                   A DUMMY FOR THE STRAIGHT FIT.                       13SEPT    14
C             NFP - I - THE NUMBER OF FIT PARAMETERS; THE NUMBER OF     BUILDF8   12
C                   COEFFICIENTS FOR A LEAST SQUARES FIT AND THE NUMBER BUILDF8   13
C                   OF POINTS FOR A SPLINE FIT.                         BUILDF8   14
C                   NUMBER OF DEGREE OF FIT FOR STRAIGHT.               13SEPT    15
C              IV - I - THE USER SUBMITTED INDEPENDENT VARIABLE.        BUILDF8   15
C         NPOINTS - I - THE NUMBER OF POINTS ON THE PLOT.               BUILDF8   16
C         IFITPTR - O - THE POINTER TO THE FIT BEAD.                    BUILDF8   17
C                                                                       BUILDF8   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/  2
      COMMON/DISPLAY/IDSTOP,IDDAD(80),NN,NNEND(10),IDSPEC(19),IDCF(41),  /DISPLAY/  2
     X               IDAXIS(4),IDTITLE(4),IDCO(23),IDMENU(20),         6JULY77    2
     X               IDINSTR(19),IDDATAB(20,4,2),NCIP,IDBLOCK(20),NCIPM 1SEPT     1
      COMMON/FITBEAD/IFTYPE,IFIVVFP,IFIVVLP,IFIVI,IFNPARA               14APR76    1
      COMMON/POINTS/X(248),Y(248),NPEC(248)                             /POINTS/   2
C                                                                       BUILDF8   23
C         GET A FIT BEAD. THE LENGTH OF THE FIT                         BUILDF8   24
C         BEAD DEPENDS UPON THE TYPE OF FIT.                            BUILDF8   25
C                                                                       BUILDF8   26
      IF(ITYPE .EQ. 10HLEASTSQUAR) CALL DMGTBD(NFP + 5,IFITPTR)         BUILDF8   27
      IF(ITYPE .EQ. 6HSPLINE) CALL DMGTBD(3 * NFP + 5,IFITPTR)          BUILDF8   28
      IF (ITYPE.EQ.6HAITKEN) CALL DMGTBD (2*NPOINTS+6,IFITPTR)          20SEPT    63
C                                                                       BUILDF8   29
C         STORE THE FIT TYPE.                                           BUILDF8   30
C                                                                       BUILDF8   31
      CALL DMSET(IFTYPE,IFITPTR,ITYPE)                                  BUILDF8   32
C                                                                       BUILDF8   33
C         STORE THE INTERVAL AND INCREMENT INFORMATION.                 14APR76   62
C                                                                       BUILDF8   35
      IF(IV .EQ. 1HY) GO TO 10                                          BUILDF8   36
      CALL DMSET(IFIVVFP,IFITPTR,X(1))                                  BUILDF8   37
      CALL DMSET(IFIVVLP,IFITPTR,X(NPOINTS))                            BUILDF8   38
      CALL DMSET(IFIVI,IFITPTR,XINCR/20.)                               BUILDF8   39
      GO TO 20                                                          BUILDF8   40
   10 CALL DMSET(IFIVVFP,IFITPTR,Y(1))                                  BUILDF8   41
      CALL DMSET(IFIVVLP,IFITPTR,Y(NPOINTS))                            BUILDF8   42
      CALL DMSET(IFIVI,IFITPTR,YINCR/20.)                               BUILDF8   43
C                                                                       BUILDF8   44
C         STORE THE NUMBER OF FIT PARAMETERS                            BUILDF8   45
C         AND THE ARRAY OF FIT PARAMETERS.                              BUILDF8   46
C                                                                       BUILDF8   47
   20 IF (ITYPE.EQ.6HAITKEN) GO TO 30                                   20SEPT    64
      CALL DMSET (IFNPARA,IFITPTR,NFP)                                  20SEPT    65
      KOMPCOD = 2000000000000B + SHIFT(NFP,18)                          14APR76   63
      CALL DMSET(KOMPCOD,IFITPTR,A)                                     BUILDF8   50
C                                                                       BUILDF8   51
```

```
      C        A SPLINE FIT IS NOT UNIQUELY DEFINED UNLESS THE POINTS ARRAYS  BUILDFB    52
      C        ARE KNOWN. ACCORDINGLY, IF THE FIT IS A SPLINE, STORE THE      BUILDFB    53
   60 C        X-COORDINATE ARRAY AND THE Y-COORDINATE ARRAY IN THE FIT HEAD  BUILDFB    54
      C        IN ADDITION TO THE ARRAY OF FIT PARAMETERS. OTHERWISE, STORE   BUILDFB    55
      C        ONLY THE ARRAY OF FIT PARAMETERS.                              BUILDFB    56
      C                                                                       BUILDFB    57
            IF(ITYPE .NE. 6HSPLINE) RETURN                                    BUILDFB    58
   65 C                                                                       BUILDFB    59
      C        STORE THE X-COORDINATE ARRAY.                                  BUILDFB    60
      C                                                                       BUILDFB    61
            KOMPCOD = KOMPCOD + NFP                                           BUILDFB    62
            CALL DMSET(KOMPCOD,IFITPTR,X)                                     BUILDFB    63
   70 C                                                                       BUILDFB    64
      C        STORE THE Y-COORDINATE ARRAY.                                  BUILDFB    65
      C                                                                       BUILDFB    66
            KOMPCOD = KOMPCOD + NFP                                           BUILDFB    67
            CALL DMSET(KOMPCOD,IFITPTR,Y)                                     BUILDFB    68
   75       RETURN                                                            BUILDFB    69
      C                                                                       20SEPT     66
      C        AITKEN FIT -- STORE NPOINTS, X AND Y ARRAYS                    20SEPT     67
      C                                                                       20SEPT     68
         30 CALL DMSET (IFNPARA,IFITPTR,NPOINTS)                              20SEPT     69
   80       CALL DMSET (IFNPARA+1,IFITPTR,NFP)                               20SEPT     70
            KOMPCOD = 700000000078+SHIFT(NPOINTS,18)                         20SEPT     71
            CALL DMSET (KOMPCOD,IFITPTR,X)                                   20SEPT     72
            KOMPCOD=KOMPCOD+NPOINTS                                          20SEPT     73
            CALL DMSET (KOMPCOD,IFITPTR,Y)                                   20SEPT     74
   85       RETURN                                                           20SEPT     75
            END                                                              BUILDFB    70
```

```
      SUBROUTINE ITRIMR(IBEAD)                                          ITRIMR     2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                /ITRIMT/    2
      COMMON/DISKT/I801,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**10  DMSET(ICOM7+J,I80A,DATA)  STORES DATA(N) IN I80A(1)    /DISKT/    3
      COMMON/OPTION/IOPTION,ISHINE                                      LASTBUX    1
      CALL DMGET(ICOM1+2,I80AD,ITRIM)                                   ITRIMR     5
      IF(ITRIM.EQ.0)GO TO 10                                            ITRIMR     6
      IF (IOPTION.NE.4HTRIM) GO TO 10                                   3AUG      16
      INEXT=2                                                           ITRIMR     7
      IPREV=4                                                           ITRIMR     8
C                                                                       ITRIMR     9
C     TRIM MODE                                                         ITRIMR    10
      RETURN                                                            ITRIMR    11
   10 CONTINUE                                                          ITRIMR    12
C     CROSSPLOT MODE                                                    ITRIMR    13
      INEXT=1                                                           ITRIMR    14
      IPREV=3                                                           ITRIMR    15
      END                                                               ITRIMR    16
```

```
1                SUBROUTINE PRNTPLT(IPLTPTR)                                PRNTPLT     2
                 COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                         /ITRIMT/    2
                 EXTERNAL SUBPLT                                            PRNTPLT     4
                 IF(IPLTPTR.EQ.0)RETURN                                     PRNTPLT     5
5                WRITE(N6,10)IPLTPTR                                        PRNTPLT     6
              10 FORMAT(1H1,50X,*LISTING OF P-M GROUP *,I10/)               PRNTPLT     7
                 CALL ITRIMR(IPLTPTR)                                       PRNTPLT     8
                 CALL SPLICE(IPLTPTR,SUBPLT, 12,1,INEXT,IPREV,0)            5OCT      106
                 END                                                        PRNTPLT    10
```

```
      SUBROUTINE SUBPLT(IBEAD1,IBEAD,IVALU)                                  PRNTPLT   13
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/    3
      COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/   2
      COMMON /TTEMP/TEMP(10)                                                 /TTEMP/    2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                     /ITRIMT/   2
      EXTERNAL SUBCRV                                                        PRNTPLT   18
      JJ=14*2**18                                                            PRNTPLT   19
      CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                        PRNTPLT   20
      WRITE(N6,10)IBEAD                                                      PRNTPLT   21
   10 FORMAT(1H0,2X ,*LISTING OF PLOT *,I10)                                 PRNTPLT   22
      WRITE(N6,20)(NREC(I),I=1,14)                                           PRNTPLT   23
   20 FORMAT(1H ,12I8,1X,A10,1X,F10.5)                                       PRNTPLT   24
      ITITLE=NREC(6)                                                         PRNTPLT   25
      CALL PRNTEXT(ITITLE,10H     TITLE)                                     PRNTPLT   26
      JJ=7*2**18                                                             PRNTPLT   27
      IF(NREC(5).EQ.0)GO TO 45                                               PRNTPLT   28
      CALL DMGET(ICOM7+JJ,NREC(5),TEMP)                                      PRNTPLT   29
      WRITE(N6,30)NREC(5)                                                    PRNTPLT   30
   30 FORMAT(1H0,6X ,*LISTING OF PLOT GROUP *,I10)                           PRNTPLT   31
      WRITE(N6,40)(TEMP(I),I=1,7)                                            PRNTPLT   32
   40 FORMAT(1H ,7I8)                                                        PRNTPLT   33
      CALL PRNTEXT(TEMP(5),10H CONDITION)                                    PRNTPLT   34
   45 CONTINUE                                                               PRNTPLT   35
      IF(NREC(7).EQ.0)GO TO 65                                               PRNTPLT   36
      WRITE(N6,50)NREC(7)                                                    PRNTPLT   37
   50 FORMAT(1H0,6X ,*LISTING OF AXIS *,I10)                                 PRNTPLT   38
      CALL DMGET(ICOM7+JJ,NREC(7),TEMP)                                      PRNTPLT   39
      WRITE(N6,60)(TEMP(I),I=1,7)                                            PRNTPLT   40
   60 FORMAT(1H ,4I8,1X,A10,I5,1X,A10)                                       PRNTPLT   41
      CALL PRNTEXT(TEMP(3),10H       AXIS)                                   PRNTPLT   42
   65 CONTINUE                                                               PRNTPLT   43
      IF(NREC(9).EQ.0)GO TO 75                                               PRNTPLT   44
      WRITE(N6,70)NREC(9)                                                    PRNTPLT   45
   70 FORMAT(1H0,6X ,*LISTING OF P-M *,I10)                                  PRNTPLT   46
      CALL DMGET(ICOM7+JJ,NREC(9),TEMP)                                      PRNTPLT   47
      WRITE(N6,40)(TEMP(I),I=1,7)                                            PRNTPLT   48
      PARAM = TEMP(6)                                                        PRNTPLT   49
      TMULT = TEMP(7)                                                        PRNTPLT   50
   40 CALL PRNTEXT(PARAM,10H PARAMETER)                                      PRNTPLT   51
      CALL PRNTEXT(TMULT,10HMULTIPLIER)                                      PRNTPLT   52
   75 CONTINUE                                                               PRNTPLT   53
      IF(NREC(8).EQ.0)GO TO 95                                               PRNTPLT   54
      WRITE(N6,80)NREC(8)                                                    PRNTPLT   55
   80 FORMAT(1H0,6X ,*LISTING OF AXIS SCALE *,I10)                           PRNTPLT   56
      JJ=8*2**18                                                             T30AUG    23
      CALL DMGET(ICOM7+JJ,NREC(8),TEMP)                                      PRNTPLT   58
      WRITE (N6,90) (TEMP(I),I=1,8)                                          T30AUG    24
   90 FORMAT (1H ,5X,4G16.8)                                                 22JUNE    83
   95 CONTINUE                                                               PRNTPLT   61
      IF(NREC(11).EQ.0)GO TO 99                                              PRNTPLT   62
      CALL SPLTOE2(NREC(11),SUBCRV,2,0,1,2,0)                                PRNTPLT   63
   99 CONTINUE                                                               PRNTPLT   64
      END                                                                    PRNTPLT   65
```

```
      1                SUBROUTINE SUBCRV(IREAD1,IBEAD,IVALU)                                  PRNTPLT    68
                       COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                     /ITRIMT/     2
                       COMMON/POINTS/X(248),Y(248),NREC(248)                                  /POINTS/     2
                       COMMON /TTEMP/TEMP(10)                                                 /TTEMP/     2
      5                COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG               30AUG        1
              C        J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)         /DISKT/      3
                       WRITE(N6,10)IBEAD                                                      PRNTPLT     73
                10     FORMAT(1HC,11X,*LISTING OF CURVE *,I10)                                PRNTPLT     74
                       JJ=12*2**18                                                            PRNTPLT     75
     10                CALL DMGET(ICOM7+JJ,IBEAD,NREC)                                        PRNTPLT     76
                       IF(NREC(10).NE.0)GO TO 15                                              PRNTPLT     77
                       WRITE(N6,19)(NREC(I),I=1,12)                                           PRNTPLT     78
                19     FORMAT(1H ,11X,2I8,1X,A10,1X,G16.8,I5,I3,3I8,1X,I2 ,1X,I8,1X,I10)      22JUNE      84
                       GO TO 16                                                               PRNTPLT     80
     15                15     WRITE(N6,20)(NREC(I),I=1,12)                                   PRNTPLT     81
                16     CONTINUE                                                               PRNTPLT     82
                20     FORMAT(1H ,11X,2I8,1X,A10,1X,G16.8,I5,I3,3I8,1X,A2 ,1X,I8,1X,A10)      22JUNE      85
                       IR=NREC(7)                                                             PRNTPLT     84
                       IX=NREC(8)                                                             PRNTPLT     85
     20                IY=NREC(9)                                                             PRNTPLT     86
                       NPTS=NREC(5)                                                           PRNTPLT     87
                       IFIT=NREC(11)                                                          PRNTPLT     88
                       JJ=NPTS*2**18                                                          PRNTPLT     89
                       IF(IX.EQ.0)GO TO 35                                                    PRNTPLT     90
     25                CALL DMGET(ICOM7+JJ,IR,NREC)                                           PRNTPLT     91
                       CALL DMGET(ICOM7+JJ,IX,X)                                              PRNTPLT     92
                       CALL DMGET(ICOM7+JJ,IY,Y)                                              PRNTPLT
                       WRITE(N6,30)                                                           PRNTPLT
                30     FORMAT(1HC,18X,*REC*,17X,1HX,19X,1HY)                                  22JUNE      96
     30                WRITE(N6,40)(NREC(I),X(I),Y(I),I=1,NPTS)                               PRNTPLT     96
                40     FORMAT(1H ,11X,I10,10X,G16.8,4X,G16.8)                                 22JUNE      97
                35     CONTINUE                                                               PRNTPLT     98
                       IF(IFIT.LE.0)RETURN                                                    PRNTPLT     99
                       CALL DMGET(ICOM1+1,IFIT,FTYPE)                                         PRNTPLT    100
     35                CALL DMGET(ICOM1+2,IFIT,XO)                                            PRNTPLT    101
                       CALL DMGET(ICOM1+3,IFIT,XN)                                            PRNTPLT    102
                       CALL DMGET(ICOM1+4,IFIT,DELX)                                          PRNTPLT    103
                       IF(FTYPE.EQ.6HAITKEN)GO TO 205                                         TOM         29
                       CALL DMGET(ICOM1+5,IFIT,NP)                                            PRNTPLT    104
     40                JJ=NP*2**18+5                                                          PRNTPLT    105
                       CALL DMGET(ICOM7+JJ,IFIT,X)                                            PRNTPLT    106
                       WRITE(N6,100)FTYPE,XO,XN,DELX                                          PRNTPLT    107
                100    FORMAT(1H0,15X,A10,* FIT *,3F16.5)                                     PRNTPLT    108
                       WRITE(N6,101)NP,(X(I),I=1,NP)                                          PRNTPLT    109
     45                101    FORMAT(1HC,20X,I5,* PARAMETERS ARE */(30X,5E16.8))             PRNTPLT    110
                       RETURN                                                                 TOM         30
                205    CALL DMGET(ICOM1+6,IFIT,IDEG)                                          TOM         31
                       WRITE (N6,100) FTYPE,XO,XN,DELX                                        27MAY        8
                       WRITE(N6,105)IDEG                                                      TOM         33
     50                105    FORMAT(1H ,15X,*DEGREE  *,I2)                                  TOM         34
                       END                                                                    PRNTPLT    111
```

```
      SUBROUTINE PRNTEXT(IBEAD,TNAME)                                    PRNTEXT     2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG           30AUG       1
C     J=N*2**15  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)     /DISKT/     3
      COMMON /TTEMP/TEMP(10)                                             /TTEMP/     2
      COMMON/ITRIMT/ITRIM,INEXT,IPREV,N6                                 /ITRIMT/    2
      INTEGER TEMP                                                       PRNTEXT     6
      IBD=IBEAD                                                          PRNTEXT     7
      JJ=10*2**15                                                        PRNTEXT     8
   99 IF(IBD.LE.0)RETURN                                                 PRNTEXT     9
      CALL DMGET(ICOM7+JJ,IBD,TEMP)                                      PRNTEXT    10
      NW=TEMP(2)/10                                                      PRNTEXT    11
      IF(( TEMP(2)-10*NW).GT.0)NW=NW+1                                   PRNTEXT    12
      NW=NW+2                                                            PRNTEXT    13
      WRITE(N6,100)TNAME,IBD,(TEMP(I),I=1,NW)                            PRNTEXT    14
  100 FORMAT(1H ,30X,A10,* TEXT *,I10/1H ,15X,2I10,1X,8A10)              PRNTEXT    15
      IBD=TEMP(1)                                                        PRNTEXT    16
      GO TO 99                                                           PRNTEXT    17
      END                                                                PRNTEXT    18
```

```
      SUBROUTINE MRCLEAN(NPOINTS)                                          MRCLEAN    2
C                                                                          MRCLEAN    3
C     THIS ROUTINE CLEANS UP THE CORE RESIDENT POINTS ARRAYS BY            MRCLEAN    4
C     REMOVING INACTIVE POINTS AND POINTS WHICH ARE OFF SCALE FROM         MRCLEAN    5
C     THE IN-CORE POINTS ARRAYS. PRIMARY INPUTS TO THE ROUTINE ARE         MRCLEAN    6
C     THE POINTS ARRAYS, THE STATUS ARRAY, AND THE SCALE INFORMATION       MRCLEAN    7
C     THE PRIMARY OUTPUT FROM THE ROUTINE IS SANITIZED POINTS ARRAYS       MRCLEAN    8
C     WHICH ARE READY TO BE FED TO THE CURVE FITTING ROUTINES.             MRCLEAN    9
C     NOTE---THE DATA MANAGER FILE CONTINUES TO HOLD THE                   MRCLEAN   10
C           UNSANITIZED ARRAYS.                                            MRCLEAN   11
C                                                                          MRCLEAN   12
C     NPOINTS - I/O - WHEN NPOINTS IS INPUT, IT IS EQUAL TO THE            MRCLEAN   13
C               NUMBER OF POINTS IN THE COORDINATE BEADS. WHEN             MRCLEAN   14
C               NPOINTS IS OUTPUT, IT IS EQUAL TO THE NUMBER OF            MRCLEAN   15
C               POINTS WHICH ARE DISPLAYED ON THE SCREEN. THE              MRCLEAN   16
C               TWO ARE NOT NECESSARILY THE SAME.                          MRCLEAN   17
C                                                                          MRCLEAN   18
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                       /AXSCALE/  2
      COMMON/POINTS/X(248),Y(248),NREC(248)                                /POINTS/   2
C                                                                          MRCLEAN   21
C     NPNOP --- THE NUMBER OF POINTS NOT ON THE PLOT.                      MRCLEAN   22
C                                                                          MRCLEAN   23
      NPNOP = 0                                                            MRCLEAN   24
      DO 20 I=1,NPOINTS                                                    MRCLEAN   25
      IF(NREC(I) .LT. 0) GO TO 10                                          MRCLEAN   26
      IF( (X(I) .LT. XMIN) .OR. (X(I) .GT. XMAX) ) GO TO 10                MRCLEAN   27
      IF( (Y(I) .LT. YMIN) .OR. (Y(I) .GT. YMAX) ) GO TO 10                MRCLEAN   28
C                                                                          MRCLEAN   29
C     THE "I-TH" POINT IS ON THE PLOT(CURRENTLY DISPLAYED ON THE           MRCLEAN   30
C     SCREEN). IF ANY POINTS PRIOR TO THE "I-TH" POINT ARE NOT ON          MRCLEAN   31
C     THE PLOT, THEN MOVE THE "I-TH" POINT INTO THE GAP CREATED            MRCLEAN   32
C     BY THE MISSING POINT(S). OTHERWISE, GO TO THE "(I+1)-ST" POINT       MRCLEAN   33
C                                                                          MRCLEAN   34
      IF(NPNOP .EQ. 0) GO TO 20                                            MRCLEAN   35
      X(I-NPNOP) = X(I)                                                    MRCLEAN   36
      Y(I-NPNOP) = Y(I)                                                    MRCLEAN   37
      GO TO 20                                                             MRCLEAN   38
C                                                                          MRCLEAN   39
C     THE POINT IS NOT ON THE PLOT.                                        MRCLEAN   40
C                                                                          MRCLEAN   41
   10 NPNOP = NPNOP + 1                                                    MRCLEAN   42
   20 CONTINUE                                                             MRCLEAN   43
      NPOINTS =NPOINTS - NPNOP                                             MRCLEAN   44
      RETURN                                                               MRCLEAN   45
      END                                                                  MRCLEAN   46
```

```
      SUBROUTINE PPLOT(IBEAD1,IBEAD,IVALU)                              PPLOT      2
      COMMON/DISKT/IBD1,ICOM1,ICOM7,XVAL,XX(20),YY(20),IC,NDEG          30AUG      1
C     J=N*2**18  DMSET(ICOM7+J,IBDA,DATA)  STORES DATA(N) IN IBDA(1)    /DISKT/    3
      COMMON/AXSCALE/XMIN,XINCR,XMAX,YMIN,YINCR,YMAX                    /AXSCALE/  2
      EXTERNAL PCURVE2                                                  PPLOT      5
      CALL DMGET(ICOM1+8,IBEAD,IAX)                                     PPLOT      6
      CALL DMGET(ICOM1+2,IAX,XINCR)                                     PPLOT      7
      CALL DMGET(ICOM1+4,IAX,YINCR)                                     PPLOT      8
      CALL DMGET(ICOM1+11,IBEAD,ICRV1)                                  PPLOT      9
      CALL SPLINE2 (ICRV1,PCURVE2,2,0,1,2,0)                            PPLOT     10
      END                                                               PPLOT     11
```

REFERENCES

1. UPDATE REFERENCE MANUAL, CDC Publication Number 60342500, REVISION G, 1975.

2. 274 INTERACTIVE GRAPHICS REFERENCE MANUAL, CDC Publication Number 60358800, 1974.

3. FORTRAN EXTENDED VERSION 4 REFERENCE MANUAL, CDC Publication Number 60497800, April 1977.

4. CYBER CONTROL LANGUAGE, ASD Computer Center, Wright-Patterson Air Force Base, Ohio, December 1977.

*U.S.Government Printing Office: 1978 — 757-080/772